

INSTITUTO NACIONAL DE METEOROLOGIA

SERVICIO DE PREDICCIÓN NUMÉRICA	NOTA TÉCNICA Nº 30	Rev. 1: 7/7/93
---------------------------------------	---------------------------	----------------

**PRUEBAS PRELIMINARES
DEL MODELO DE
ADAPTACION DE VIENTOS DE
MASS Y DEMPSEY**



20 JUN 2001

AEMET-BIBLIOTECA



1013560

José Ardao Berdejo

PRUEBAS PRELIMINARES DEL MODELO DE ADAPTACIÓN DE VIENTOS DE MASS Y DEMPSEY

Autor: José Ardao Berdejo

- 1.- Introducción**
- 2.- Descripción del modelo**
- 3.- Procedimientos de ejecución**
 - 3.1.- Obtención de datos**
 - 3.2.- Descripción del código**
 - 3.3.- Opciones del NAMELIST**
- 4.- Descripción del trabajo realizado**
- 5.- Conclusiones**

Apéndice A (Bibliografía)

Apéndice B (Casos del Estrecho y del PYREX)

Apéndice C (Código del MASS, última versión)

1.- Introducción

El presente documento es el resultado de los tres meses de prácticas, del autor, en el Servicio de Predicción Numérica del Instituto Nacional de Meteorología. La finalidad perseguida en el citado período de prácticas, fue la de realizar las pruebas previas a la verificación del modelo para mostrar tanto los aspectos potencialmente positivos como los negativos, de dicho modelo. Por otra parte, en el presente documento se pretende además describir brevemente el modelo y, con algo más de profundidad, el modo de operar para ejecutarlo, así como los archivos de que consta.

La codificación del modelo fue realizada por Ernesto Rodríguez Camino, durante el año 1990. El autor de este documento se limitó básicamente a modificar la subrutina que produce las salidas gráficas, ejecutar el modelo en varias situaciones y redactar el presente documento.

Antes de entrar en la descripción del modelo es conveniente aclarar que se trata de un modelo de adaptación de vientos, esto quiere decir que no se persigue una modelización exhaustiva de la atmósfera, lo que se busca es obtener unos resultados aceptables de un campo concreto, como es el viento, utilizando modestos recursos informáticos y con poco tiempo de cálculo. Así, la formulación matemática se reduce en realidad a dos ecuaciones: la de la cantidad de movimiento y la de la termodinámica. Este hecho ha de ser tenido siempre en cuenta para presuponer las limitaciones del modelo y no esperar por lo tanto unos resultados excepcionales. Además el modelo no va a necesitar ni un gran ordenador para ser ejecutado, ni presenta grandes consumos de tiempo de cálculo, al contrario: con un simple ordenador personal y un tiempo del orden del minuto es bastante. Estos dos hechos nos muestran la mejor cualidad del modelo y por consiguiente cuales pueden ser sus usos: un primer uso es la formación y otro su utilización en los GPV's. En cuanto a la formación se trata de un puente ideal para quien ha de introducirse en la predicción numérica, pues reúne las características básicas de un modelo operativo, pero permitiendo al programador manejar con comodidad el código e interactuar con éste modificándolo, así como ejecutarlo en uno o pocos minutos, comprobando en tiempo real el resultado de las modificaciones. Su uso como herramienta predictiva no debe ser desdeñado, pues aún siendo un modelo limitado puede ser "sintonizado" para el área que cubre el GPV e incluso para cada situación y/o época del año, produciendo predicciones aceptables. Esto no es más que una posibilidad futura que no está hoy en día desarrollada en absoluto, pero que debido a los modestos recursos que precisa bien podría llegar a ser una realidad.

2.- Descripción del modelo

A continuación vamos a reproducir el abstract del artículo original de Mass y Dempsey en que se basó la codificación:

El artículo describe un modelo mesoescalar de un nivel, en coordenadas sigma, válido para diagnosticar el viento en superficie en zonas costeras o montañosas. El modelo precisa escasos recursos informáticos y necesita pocos datos para la inicialización. Las ecuaciones de conservación de la energía y del momento son integradas, con una altura de escala sinóptica fija y campos de temperatura estacionarios, hasta alcanzar un estado estacionario que diagnostica el viento en superficie y los campos de temperatura forzados por el terreno y sus obstáculos. Si se desea un forzamiento debido a las transferencias de calor, el modelo utiliza el estado estacionario final como si fuera un estado inicial, a partir del cual se integra el modelo incluyéndose las transferencias de calor hasta alcanzar el tiempo de verificación. El modelo no conserva la masa, pero bajo la hipótesis de flujo hidrostático el campo de masa (y por consiguiente la presión en superficie) está determinado por la estructura vertical de la temperatura, que está parametrizada en el modelo en términos de la temperatura en superficie.

En el artículo se presentan cuatro ejecuciones y sus correspondientes campos de viento observados, que apuntan la posibilidad de que el modelo pueda diagnosticar muchos de los detalles del flujo mesoescalar en terreno accidentado para varias direcciones del flujo y forzamientos de las transferencias de calor. Parece ser que el calentamiento y enfriamiento adiabático juegan un papel crucial a la hora de generar canalizaciones y variaciones en la dirección debido a la orografía. También se dan recomendaciones para posibles mejoras del modelo.

Además de la información que proporciona el abstract vamos a dar una nota extra que esperamos completen la visión del modelo. El modelo integrará la ecuación de la cantidad de movimiento, V_s , y la ecuación de la tendencia de la temperatura superficial, T_s . La expresión de estas ecuaciones en coordenadas sigma es la siguiente:

$$\frac{\partial V_s}{\partial t} = -V_s \cdot \nabla_\sigma - f k \wedge V_s - (g \nabla_\sigma z_s + R T_s \nabla_\sigma \ln p_s) + F + K_M \nabla_\sigma^2 V_s$$

y

$$\frac{\partial T_s}{\partial t} = -V_s \cdot \nabla_\sigma T_s + \frac{R T_s}{c_p} \left(\frac{\partial \ln p_s}{\partial t} + V_s \cdot \nabla_\sigma \ln p_s \right) + \frac{Q}{c_p} + K_T \nabla_H^2 T$$

donde los sumandos $g \nabla_\sigma z_s + R T_s \nabla_\sigma \ln p_s$ de la primera ecuación y $\frac{\partial \ln p_s}{\partial t} + V_s \cdot \nabla_\sigma \ln p_s$

de la segunda se pueden expresar en función de variables y constantes conocidas, básicamente temperaturas a distintos niveles.

Para un mayor detalle en la descripción del modelo lo más apropiado es consultar directamente el artículo original.

3.- Procedimientos de ejecución

A continuación vamos a describir de que ficheros consta el modelo, sus entradas y salidas, su finalidad y el modo de ejecutarlos. En el momento de redactar este documento (verano de 1993), todos los programas que componen el modelo y los programas auxiliares, están codificados de forma que funcionan en el entorno en que han sido creados, por lo que cualquier intento de instalarlos en otras máquinas requerirá forzosamente realizar modificaciones, directamente en el código, para adaptarlo a otras configuraciones.

3.1.- Obtención de datos

Los datos de orografía, temperatura y viento precisos para ejecutar el modelo van a ser distintos según el área y el período de nuestro interés, por lo que tanto si trabajamos con datos en tiempo real como si trabajamos a tiempo pasado, tendremos que obtenerlos con anterioridad a la ejecución del modelo. Para obtener estos datos disponemos hoy en día de dos programas en el Fujitsu: PREMASS y JPREMASS. El primero de ellos: PREMASS, se va a encargar de gestionar los datos, mientras que el segundo: JPREMASS, contiene las JCL's necesarias para ejecutarlo.

El programa PREMASS utiliza como entrada tanto los análisis como los previstos del modelo LAM, pudiéndose usar tanto en tiempo real como con tiempo pasado. Debido a que el modelo LAM tiene una resolución de 0.91° mientras que el modelo MASS trabaja a $10'$, la primera tarea que ha de realizar PREMASS será interpolar los campos leídos del LAM a una resolución de $10'$. Una vez interpolados los campos el programa los deposita en veinte ficheros quedando listos para su migración a un PC. El propio programa genera también un batch que se encarga de realizar esta migración, nuestra única actividad se remite a llevar este fichero batch (que además se llama BATCH), desde el Fujitsu al PC.

Como recordatorio al posible usuario, decirle que antes de ejecutar JPREMASS es necesario borrar los dos ficheros que crea este programa, pues si se los encuentra ya creados su ejecución no finalizará con éxito (estos ficheros son "PC" y "BATCH").

Otra de las características del programa PREMASS es la referente al área peninsular que nos interese. Para seleccionar un área cualquiera lo único que tenemos que hacer es darle valores, dentro del programa, a cuatro variables, esto podría haberse hecho en un fichero aparte para evitar compilar de nuevo el programa cada vez que se cambia el área, queda por tanto como una puerta abierta para futuras mejoras. Las variables en cuestión son NFIL y NCOL que nos ayudarán a fijar las esquinas Norte y Oeste del área, para decidir los valores correctos de estas variables hay que ver a que valores de latitud y longitud corresponden, esto se ve con las fórmulas que relacionan la latitud y la longitud con NFIL y NCOL en el programa. Las dos variables restantes son I y J, que aparecen en varias partes del programa y nos dan el número de puntos zonales y meridionales del rectángulo y por consiguiente determinan finalmente el área. Conviene puntualizar que el número de puntos zonales y meridionales aparecen en más sitios y no siempre con

las variables I y J. Por ejemplo aparecen en las sentencias Dimension y en llamadas a subrutinas, cuando cambiemos la ventana será necesario cambiar éstos valores.

3.2.- Descripción del código

La codificación del modelo de Mass y Dempsey se realizó por Ernesto Rodríguez Camino en lenguaje fortran. Como ya se ha mencionado este programa se encuentra en un PC y los requisitos para ejecutarlo, al menos en su estado actual, son: el fortran NDP y un PC de ciertas prestaciones. En realidad este programa se encuentra en tres versiones distintas y sucesivas, conteniendo cada una pequeñas modificaciones sobre el anterior. Las tres versiones del programa se encuentran en los ficheros: MASS.FOR, MAS.FOR y MIS.FOR. Para ejecutar uno de estos programas es preciso obtener un ejecutable, a lo que se llega del siguiente modo (en fortran NDP):

1.- BIG *nombredefichero*

2.- FASTLINK @*nombredefichero*.LNK

3.- RUN386 *nombredefichero*.exp

Con el último paso ejecutamos el programa y nos aparecerá su salida en pantalla, esto es el mapa del área seleccionada y la representación gráfica del viento.

El código completo del programa, en su última versión, se encuentra en el apéndice D, pero para bosquejar unas primeras líneas de su estructura vamos a describir someramente su contenido. Empecemos con la parte del modelo no predictiva:

El primer paso del programa, después de definir e inicializar variables, es calcular el parámetro de Coriolis, $f=2\Omega\sin\phi$, para cada latitud. Esto es para cada valor de J.

A continuación el programa lee los valores de ciertas variables (se insistirá sobre este punto posteriormente) en el NAMELIST, hecho lo cual lee en otro fichero los valores de la orografía.

El siguiente paso es filtrar las cotas con una variable del NAMELIST: OROMAX. Su efecto es reducir las cotas superiores a OROMAX al valor de esta variable. Visto de un modo más intuitivo consiste en convertir las montañas en mesetas. También se corrigen las cotas con un factor para darle mayor o menor peso al terreno, es la variable FACTOR del NAMELIST. Por último se asignan distintos rozamientos a los puntos de mar y a los de tierra.

Una vez realizado lo anterior se leen los valores de T_R y Z_R , temperatura y altura del nivel de referencia, de los ficheros correspondientes y que están en el directorio BASURA, a donde llegaron con el batch generado por PREMASS.

La siguiente tarea es obtener la temperatura de superficie, T_s , para lo cual tenemos dos posibilidades, realizándose la elección a través de la variable lógica TSUP del NAMELIST, mediante la fórmula: $T_s = T_R + \gamma(Z_R - Z_s)$. Si TSUP es falsa se calcula la temperatura de superficie a partir de la temperatura del nivel de referencia y de las cotas del suelo y del nivel de referencia, suponiendo que el perfil de temperaturas es el GAMMA que aparece en el NAMELIST. Si por el contrario el valor de TSUP es cierto se toma la temperatura de superficie del fichero correspondiente en el directorio BASURA.

Para obtener el viento inicial tenemos también dos opciones que elegimos con la variable lógica VGEOST del NAMELIST. Si VGEOST es falsa se leen los valores de las componentes del viento en sendos ficheros de BASURA. Si es cierta se calcula el viento inicial a través de la subrutina WINIT.

El siguiente paso del programa es leer los valores de las componentes del viento en 1.000 mb de sendos ficheros en BASURA.

La última tarea de la parte no predictiva del programa consiste en dibujar el relieve y el viento inicial, para lo cual se llama a la subrutina DIBWIN.

Pasemos ahora a ver la parte predictiva. Esta parte se ejecuta bajo el control de una variable del NAMELIST, la variable NSTEP. Esta variable controla el número de iteraciones que se realizan de los procesos que se describen a continuación.

Antes de entrar en el bucle controlado por NSTEP se definen las componentes del viento en la rejilla alternada de Arakawa, siguiendo el artículo original de Mass y Dempsey.

La primera tarea dentro del bucle es calcular la tendencia de la temperatura y la cantidad de movimiento, lo que se hace con la subrutina FORCST.

Basándose en estas tendencias se calculan los nuevos valores de la temperatura y el viento en el interior del área, mediante la subrutina TSTEP. En el contorno se calculan con la subrutina BOUND.

La salida gráfica se realiza cada diez pasos de tiempo, definiéndose antes las componentes del viento en la rejilla original. Dicha salida corre a cargo de la subrutina DIBWIN.

Cuando por fin el bucle alcanza el valor de NSTEP finaliza el programa.

El programa contiene un total de doce subrutinas y funciones, de las cuales ocho son subrutinas del modelo, mientras que las otras cuatro (tres funciones y una subrutina) han sido tomadas de los programas de ejemplo del fortran NDP. Veamos las misiones de las ocho subrutinas del modelo:

1.- Subrutina WINIT

Esta es una subrutina opcional, esto es que no necesariamente ha de usarse. Sólo en el caso de no tener los datos de vientos disponibles en un fichero o de no querer usarlos se ejecutará esta subrutina. La variable encargada de bifurcar el camino en uno u otro sentido es la variable VGEOST, de la que ya hablaremos al hacerlo del NAMELIST. La subrutina WINIT se encarga de calcular el viento inicial, necesario para ejecutar el modelo, imponiendo que existe el equilibrio geostrófico. Por tanto necesita los datos de temperatura y las alturas en cada punto de la red.

El viento calculado en esta subrutina presenta un término adicional que tiene en cuenta el rozamiento y que al ir multiplicado por la variable EPSI, podemos anularlo dándole a esta variable el valor cero, quedándonos un viento puramente geostrófico.

2.- Subrutina FORCST

La subrutina FORCST es el eje central del programa y en ella se calculan las tendencias de las dos ecuaciones que configuran el modelo de Mass y Dempsey, que como ya sabemos son :

$$\frac{\partial V_s}{\partial t} = -V_s \cdot \nabla_\sigma - f k \wedge V_s - (g \nabla_\sigma z_s + R T_s \nabla_\sigma \ln p_s) + F + K_M \nabla_\sigma^2 V_s$$

y

$$\frac{\partial T_s}{\partial t} = -V_s \cdot \nabla_\sigma T_s + \frac{R T_s}{c_p} \left(\frac{\partial \ln p_s}{\partial t} + V_s \cdot \nabla_\sigma \ln p_s \right) + \frac{Q}{c_p} + K_T \nabla_H^2 T$$

la ec. de la cantidad de movimiento y la de la termodinámica, ya descritas en el epígrafe segundo: Descripción del modelo.

En esta subrutina se calculan además todas las constantes necesarias para evaluar las dos ecuaciones anteriores.

3.- Subrutina BOUND

Como la subrutina anterior calcula la ecuación de la cantidad de movimiento y la de la termodinámica en todo el área salvo en los contornos, se nos va a presentar un problema a la hora de evaluar dichas variables en los contornos. En el interior del área los valores del viento y la temperatura se hallarán teniendo en cuenta las tendencias calculadas en la subrutina anterior, pero las tendencias anteriores no se pueden calcular en los contornos del área, pues al emplear una discretización por diferencias finitas necesitaríamos valores exteriores a la región que estudiamos para evaluar las tendencias en el contorno. Esto nos obliga a

) calcular el viento y la temperatura en el contorno de un modo distinto, esta es la finalidad de la subrutina BOUND.

La idea consiste en dar distintos valores según sea el flujo a través del contorno. Cuando el flujo es entrante los valores de viento y temperatura no se modifican y coinciden por tanto con los iniciales. Por contra si el flujo es saliente los valores en el contorno serán los que adveccte el viento de zonas próximas. Esto es lo que se codifica en la subrutina BOUND, el modo de adjudicar los valores advectados podría ser distinto e incluso podría ser distinto el criterio utilizado en función de los flujos.

4.- Subrutina TSTEP

En la subrutina TSTEP se calculan los valores del viento y la temperatura utilizando las tendencias calculadas en la subrutina FORCST. Estas tendencias no se suman a los valores previos de viento y temperatura directamente, se aplican antes unos esquemas para atenuar el efecto de las tendencias. Los valores en el contorno se calculan como se explico en la subrutina BOUND.

5.- Subrutina LAPSIG

Esta subrutina se limita a calcular el laplaciano de un campo cualquiera. Se usa varias veces en el programa.

6.- Subrutina DIBWIN

La subrutina DIBWIN es la encargada de realizar las salidas gráficas del modelo y es donde se concentró la mayor parte de la actividad del autor. Se encarga de dibujar en distintos colores los distintos intervalos de altura del suelo, así como los símbolos del viento y algunos de los valores del NAMELIST.

7.- Subrutina IBUF

Esta subrutina es la encargada de leer los campos que precisa el modelo del directorio C:\BASURA donde se encuentran.

8.- Subrutina SMOOTH

Esta subrutina realiza el suavizado del campo entrante. Se puede anular o bien variarla desde el NAMELIST con la variable RNU.

3.3.- Opciones del NAMELIST

En el NAMELIST se encuentran las variables, y sus valores, que nos permiten controlar las opciones del modelo. Su contenido literal es:

```
&DATOS
DIA='15',      HORA='I',      ALCAN='BA',      REFLEV='85',
SLAT=37.0,     DPHI=0.1660,     KT=300000.,     KM=30000.,
RNU=0.50,      SMOROG=0.20,     FACTOR=1.0,     OROMAX=500.0,
NSTEP=50,      NADIAB=90,        DT=240.,        DUSMIN=0.000000001,
GAMMA=0.006,   TSUP=.FALSE.,     VGEOST=.FALSE.,CORIOL=.T.,
CDSEA=0.0014,  CDLAND=0.02,      CROZ=0.2,       EPSI=1.,
ACSEA=0.04,    ACLAND=1.0,       SIZE=1.5,       ANCHO=8., ISKIPP=2,
&END
$DATOS
```

veámoslo variable por variable.

La primera línea contiene las variables que se encargan de definir los ficheros del directorio BASURA, que contienen los datos que necesita el programa para ejecutarse.

Veamos que información contiene el nombre fichero: un posible nombre para un fichero de BASURA es el siguiente: BIBA8515.DAT. Todos los ficheros tienen extensión DAT. Los dos últimos dígitos del nombre de fichero corresponden al día, en este caso 15. Los dígitos quinto y sexto corresponden al nivel, p. ej. 850 mb. será 85. El tercer y cuarto dígito nos dicen de que previsto se trata, esto es el alcance, en nuestro caso BA = 12 hrs. El segundo dígito nos da la hora y el primer dígito el tipo de variable. La equivalencia entre el primer dígito y el campo correspondiente se visualiza en la siguiente tabla:

A	Geopotencial		I	Componente u
B	Temperatura		J	Componente v

.- Variable DIA: Con esta variable fijamos el día que deseamos estudiar. El valor de esta variable coincidirá con los dos últimos dígitos del nombre del fichero.

- Variable HORA: Con esta variable fijamos la hora que deseamos estudiar. El valor de esta variable ocupa el segundo dígito en el nombre del fichero. La correspondencia entre el valor de la variable y la hora que representa se puede ver en la siguiente tabla:

I	00 Z
J	06 Z
K	12 Z
L	18 Z

- Variable ALCAN: Esta variable fija el alcance que deseamos estudiar, esto es el previsto. La tabla siguiente nos da una equivalencia entre el alcance en horas y los valores usados por la variable ALCAN:

0	AA		12	BA		24	CA		36	DA		48	EA
1	AB		13	BB		25	CB		37	DB		49	EB
2	AC		14	BC		26	CC		38	DC		50	EC
3	AD		15	BD		27	CD		39	DD		52	ED
4	AE		16	BE		28	CE		40	DE		53	EE
5	AF		17	BF		29	CF		41	DF		54	EF
6	AG		18	BG		30	CG		42	DG		55	EG
7	AH		19	BH		31	CH		43	DH		56	EH
8	AI		20	BI		32	CI		44	DI		57	EI
9	AJ		21	BJ		33	CJ		45	DJ		58	EJ
10	AK		22	BK		34	CK		46	DK		59	EK
11	AL		23	BL		35	CL		47	DL		60	EL

El alcance ocupa los lugares tercero y cuarto en el nombre del fichero.

- Variable REFLEV: Esta variable nos permite especificar el nivel que nos interese.

- Variable SLAT: En SLAT aparece la latitud más meridional del área que estudiamos. Esta variable se utiliza para calcular el parámetro de Coriolis, $f=2\Omega\sin\phi$, al principio del programa, como ya se ha indicado.

- Variable DPHI: Esta variable nos da la distancia en grados de arco de círculo máximo entre dos puntos sucesivos de la malla utilizada. En nuestro caso diez minutos, que expresado en grados corresponde a 0.16667° .

- Variable KT: Esta variable corresponde al coeficiente K_T que aparece multiplicando al laplaciano de la temperatura en la ecuación de la termodinámica. Cuanto mayor sea su valor mayor difusión hay.

- Variable KM: Esta variable corresponde al coeficiente K_M que aparece multiplicando al laplaciano de la velocidad del viento en la ecuación de la cantidad de movimiento. Cuanto mayor sea el valor de este coeficiente mayor difusión habrá.

- Variable RNU: Con esta variable se calculan los dos coeficientes que aparecen en la subrutina de suavizado SMOOTH.

- Variable SMOROG: En ninguna de las tres versiones (MASS.FOR, MAS.FOR y MIS.FOR) se utiliza esta variable.

- Variable FACTOR: Esta variable aparece en el programa multiplicando a las cotas, por lo que nos permite aumentar o disminuir la orografía de un modo global.

- Variable OROMAX: La misión de esta variable es filtrar las cotas superiores a OROMAX. La función de esta variable ya ha sido comentada al describir la estructura del programa.

- Variable NSTEP: Esta variable es el número de pasos de tiempo hasta el que dejamos ejecutar el programa.

- Variable NADIAB: Las transferencias de calor (cuarto término en la ec de la termodinámica) no se tienen en cuenta hasta que el paso de tiempo supera a NADIAB. Así en nuestro ejemplo de NAMELIST, al poner NADIAB mayor que NSTEP no tendremos las transferencias de calor en cuenta.

- Variable DT: Esta variable se utiliza en la subrutina TSTEP y corresponde al paso de tiempo utilizado. En el NAMELIST que hemos dado de ejemplo aparece $DT = 240$, esto es un paso de tiempo de 240 segundos. Como el valor que aparece de NSTEP es cincuenta, resulta que estamos integrando hasta 3.3 horas, pues $240 * 50 = 12.000$ sg, que son 3.3 horas.

- Variable DUSMIN: Esta variable nos permite dejar de integrar en el tiempo cuando las diferencias de campos de vientos sucesivos son pequeñas. La magnitud de las diferencias se limita con la variable DUSMIN.

- Variable GAMMA: Esta variable representa el perfil de temperatura entre Z_R y Z_H . Correspondiéndose con la γ del artículo de MASS y DEMPSEY.

- Variable TSUP: Esta variable nos permite leer la temperatura superficial en un fichero de BASURA cuando su valor es cierto, o bien calcularla, con la fórmula:

$$T_S = T_R + \gamma(Z_R - Z_S) \text{ , cuando es falsa.}$$

- Variable VGEOST: Con esta variable leemos las dos componentes del viento inicial en sendos ficheros cuando su valor es falso, o bien las calculamos mediante la subrutina WINIT cuando es cierto.

- Variable CORIOL: Con esta variable podemos tener en cuenta o anular el término de Coriolis en la ecuación de la cantidad de movimiento. Con el valor verdadero tendremos en cuenta el término de Coriolis y con el valor falso lo anularemos.

- Variable CDSEA: Esta variable se usa para calcular un coeficiente corrector del rozamiento, $CD(I,J)$, que tomará distinto valor sobre tierra y sobre mar y que además se usará en los términos del viento que incluyan el rozamiento.

- Variable CDLAND: Esta variable hace lo mismo que la anterior. La finalidad de esta pareja de variables es establecer un rozamiento diferente para tierra y para mar.

- Variable CROZ: Esta variable se utiliza tanto en la subrutina WINIT como en la FORCST y la podemos interpretar como un coeficiente de rozamiento que multiplica al término de fricción de una ecuación, permitiéndonos darle mayor o menor peso a dicho término.

- Variable EPSI: Esta variable se usa en la subrutina WINIT y nos permite quitar el término de rozamiento a la hora de calcular el viento inicial, quedándonos con un viento puramente geostrófico.

- Variable ACSEA: Esta variable corresponde al término de calentamiento,

$$\frac{Q}{c_p} \text{ , de la ecuación de la termodinámica sobre el mar.}$$

- Variable ACLAND: Es análoga a la anterior, pero sobre tierra.

- Variable SIZE: Esta variable nos permite cambiar el tamaño de las flechas que simbolizan el viento en la salida gráfica. Se usa en la subrutina DIBWIN.

- Variable ANCHO: Esta variable nos permite cambiar el tamaño de la ventana en la que se produce la salida gráfica. También se usa en DIBWIN.

- Variable ISKIPP: Con esta variable podemos determinar cada cuantos puntos de rejilla se va a representar un símbolo del viento. También se usa en DIBWIN.

4.- Descripción del trabajo realizado

Tras las obligadas lecturas y familiarización tanto del artículo de Mass y Dempsey como del código, se realizaron varias ejecuciones del modelo y se modificaron algunas de las líneas del código. Principalmente de la subrutina encargada de las salidas gráficas.

Empezaremos por describir las ejecuciones que se presentan. Encontraremos dos bloques diferenciados: el primero corresponde al Estrecho en una situación que cambió de Poniente a Levante y el segundo corresponde al POI nº9 del experimento PYREX.

Bloque I, situación del Estrecho:

El primer conjunto de mapas corresponde a las situaciones iniciales, de los seis alcances con los que se trabajó. El resto de los mapas serán los resultados finales, para los seis alcances, bajo distintos valores de ciertas variables.

La primera serie de mapas que podemos encontrar tras la serie de situaciones iniciales, corresponde a unos valores del NAMELIST estándar, en la que se ha tratado de ver la influencia del término de Coriolis en la ecuación de la cantidad de movimiento. En la parte superior de cada hoja aparece la salida en la que se tiene en cuenta el término de Coriolis y en la parte inferior la misma salida, pero anulando el término de Coriolis.

En la tercera serie de mapas hemos repetido el caso anterior, pero reduciendo el rozamiento. Podemos volver a ver el efecto del término de Coriolis y además que efecto tiene darle al rozamiento la mitad del peso anterior.

En la cuarta serie de mapas hacemos otra vez lo mismo, pero con $CROZ = 0.2$.

En la quinta serie vemos cual es el efecto del gradiente de temperatura vertical. En la parte superior aparece $GAMMA = 0.004$ y en la inferior $GAMMA = 0.008$

Bloque II, caso del PYREX:

En el caso del PYREX cambiamos dos cosas: el viento lo representaremos con banderolas en vez de flechas y en lugar de presentar sólo la situación inicial y la final, presentaremos además toda la serie de estados intermedios.

La sexta serie contiene el estado inicial, el final y los cuatro estados intermedios en una situación estándar. El rozamiento no está modificado.

En la séptima serie repetimos el caso anterior, pero con el rozamiento disminuido en un cincuenta por ciento.

En la octava serie disminuimos aún más el rozamiento, $CROZ = 0.2$.

En la novena serie disminuimos a la mitad el valor del rozamiento en tierra, esto es: pasamos de $CDLAND = 0.02$ a $CDLAND = 0.01$, manteniendo $CROZ = 0.2$ en lugar del valor estándar que sería $CROZ = 1$.

En la décima serie volvemos a reducir el valor del rozamiento en tierra a la mitad. Esto es: $CDLAND = 0.005$.

Modificaciones en el código:

En la subrutina DIBWIN se modificó buena parte de ella. Inicialmente dibujaba los vientos como flechas, así como un círculo por cada punto de rejilla. Azul si era de mar y amarillo si era de tierra. Las modificaciones que se realizaron llevaron a las salidas que se presentan en este documento. Inicialmente se modificó la forma de presentar los puntos de tierra, empleando distintos colores (o tonos de gris en las salidas impresas) para los distintos intervalos de altura. Posteriormente se amplió el número de variables del NAMELIST que aparecen junto al mapa. Finalmente se cambió la representación del viento en forma de flechas por las banderolas usuales de meteorología.

También se realizaron cambios menores en el código que no se describen.

5.- Conclusiones

Parte de las conclusiones ya se esbozaron en la introducción, como los modestos recursos informáticos que precisa el modelo o su utilidad didáctica. Que sólo un PC y algunos minutos de cálculo sean suficientes para ejecutar el modelo es tal vez la mejor demostración de los escasos recursos informáticos que precisa el modelo y sin duda su punto más fuerte. La utilidad didáctica como puente para introducirse en la predicción numérica ya fue expuesta en la introducción.

Como ya se apuntó en el segundo punto, "descripción del modelo", las dos ecuaciones en que se basa el modelo son la de la cantidad de movimiento y la de la termodinámica, pero no se tiene en cuenta la ecuación de continuidad, o sea la conservación de la masa, aunque algunos de los modelos más antiguos de los citados en la bibliografía sí se basan en la conservación de continuidad. Cuando realizamos las pruebas que se exponen en este documento hubo un hecho que nos sorprendió: el viento zonal no se aceleraba en el Estrecho tal y como ocurre en la realidad. Esta deficiencia, debida probablemente a la no inclusión de la ecuación de continuidad, también se puede detectar, tras un estudio minucioso, en las salidas gráficas del artículo original de Mass y Dempsey. Siendo tal vez ésta la mayor limitación del modelo. Desde luego para nosotros fue el peor descubrimiento de los que hicimos.

Además de la limitación debida a la ecuación de continuidad, se nos presentan otras dos posibles limitaciones: una es el ser un modelo de un solo nivel y otra la gran cantidad de constantes empíricas de que consta. El hecho de no incluir más que un nivel le proporciona gran sencillez al modelo, pero al mismo tiempo es probablemente una simplificación excesiva. Tal vez un modelo de tres o cuatro niveles proporcionase mejores resultados. El problema de las constantes empíricas reside en el gran número de ellas que es preciso tocar para sintonizar apropiadamente el modelo. Esto nos llevaría a tener que realizar numerosos estudios para fijar sus valores, pudiéndose dar la situación de que dichas constantes tuviesen distintos valores según la época del año o la zona geográfica en la que se aplicase el modelo.

APENDICE A

(Bibliografía)

P. Alpert and B. Getenio, "One-level diagnostic modeling of mesoscale surface winds over complex terrain. Part I: Comparison with three-dimensional modeling in Israel", *Mon. Wea. Rev.*, 116, 2025-2046 (1988).

P. Alpert, B. Getenio and R. Zak-Rosenthal, "One-level diagnostic modeling of mesoscale surface winds over complex terrain. Part II: Applicability to short-range forecasting", *Mon. Wea. Rev.*, 116, 2047-2061 (1988).

M. Danard, "A simple model for mesoscale effects of topography on surface winds", *Mon. Wea. Rev.*, 105, 572-580 (1971).

M. Danard, "A prognostic model for the surface temperature, height of the atmospheric boundary layer and surface wind", *MWR*, 117, 67-77 (1989).

M.A. Estoque, "A surface mesoscale wind model for complex terrain", *Atmósfera*, 3, 203-216 (1990).

D. Heinman, "Estimation of regional surface layer windfield characteristics using a three-layer mesoscale model", *Beitr. Phys. Atmosph.*, 59, 518-537 (1986).

C.F. Mass and D.P. Dempsey, "A one-level mesoscale model for complex terrain", *Proc. Sixth Conf. on Numerical Weather Prediction, Amer. Meteor. Soc.*, 343-347 (1983).

C.F. Mass and D.P. Dempsey, "A one-level, mesoscale model for diagnosing surface winds in mountainous and coastal regions", *Mon. Wea. Rev.*, 113, 1211-1227 (1985).

E. Olsson, "Dynamic simulation of surface winds", Fou-nofisier, Workshop on simplified dynamical models for short-range forecasting on the mesoscale", 18-19 June 1984, SMHI-Report R&D notes No 37. Norkoepping, Suecia (1984).

C. Ramis, A. Jansá y S. Alonso, "Sea breeze in Mallorca. A numerical study", *Meteor. Atmos. Phys.*, 42, 249-258 (1990).

D. Rosseau and H.L. Pham, "Adaptation dynamique des previsions", Note de travail de l'Etablissement d'Etudes et de Recherches Météorologiques. No 37. Norrkoepping, Suecia (1983).

C.A. Sherman, "A mass-consistent model for wind fields over complex terrain", *J. Appl. Meteor.*, 17, 312-319 (1978).

APENDICE B
(Casos del Estrecho
y del PYREX)

BLOQUE I

SITUACION DEL ESTRECHO

1ª Serie

Situaciones Iniciales

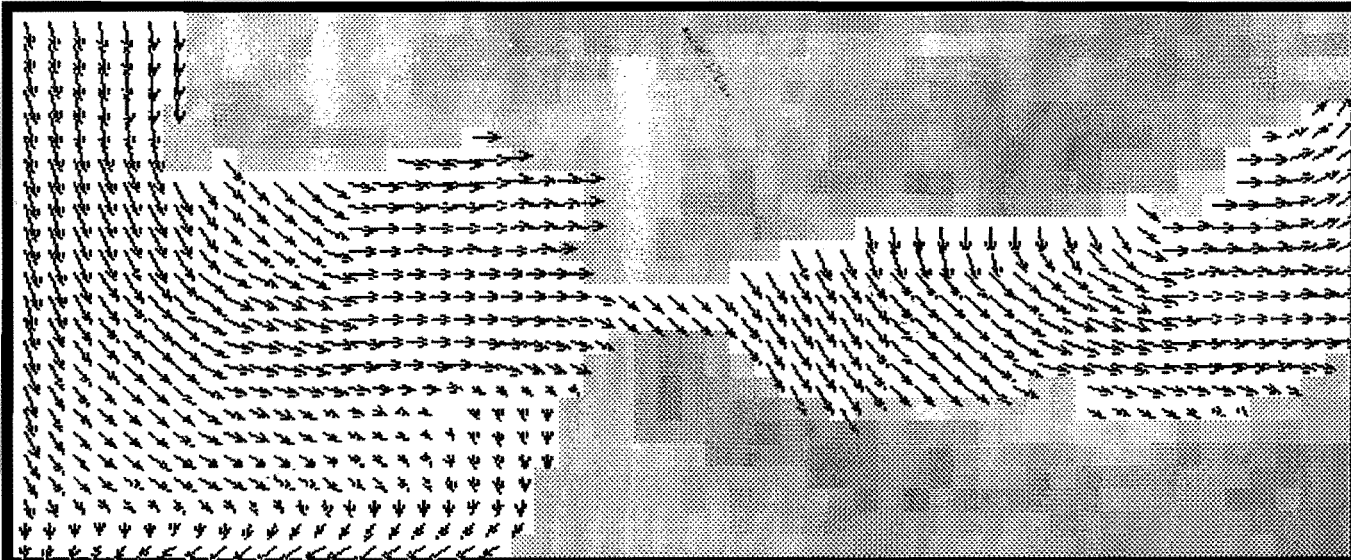
FINISHED, PRESS A KEY

WAIT A MOMENT

VIENTO INICIAL

20 M/S

0



PRESS A KEY TO CONTINUE

ALCAN = AG

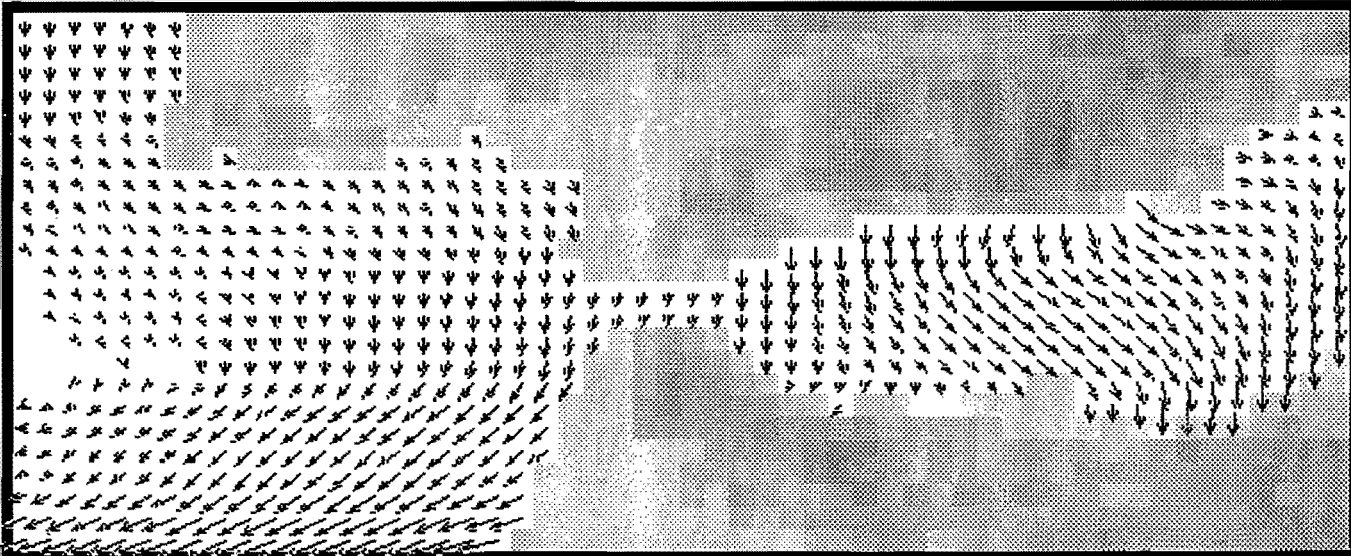
FINISHED, PRESS A KEY

WAIT A MOMENT

VIENTO INICIAL

20 M/S

0



PRESS A KEY TO CONTINUE

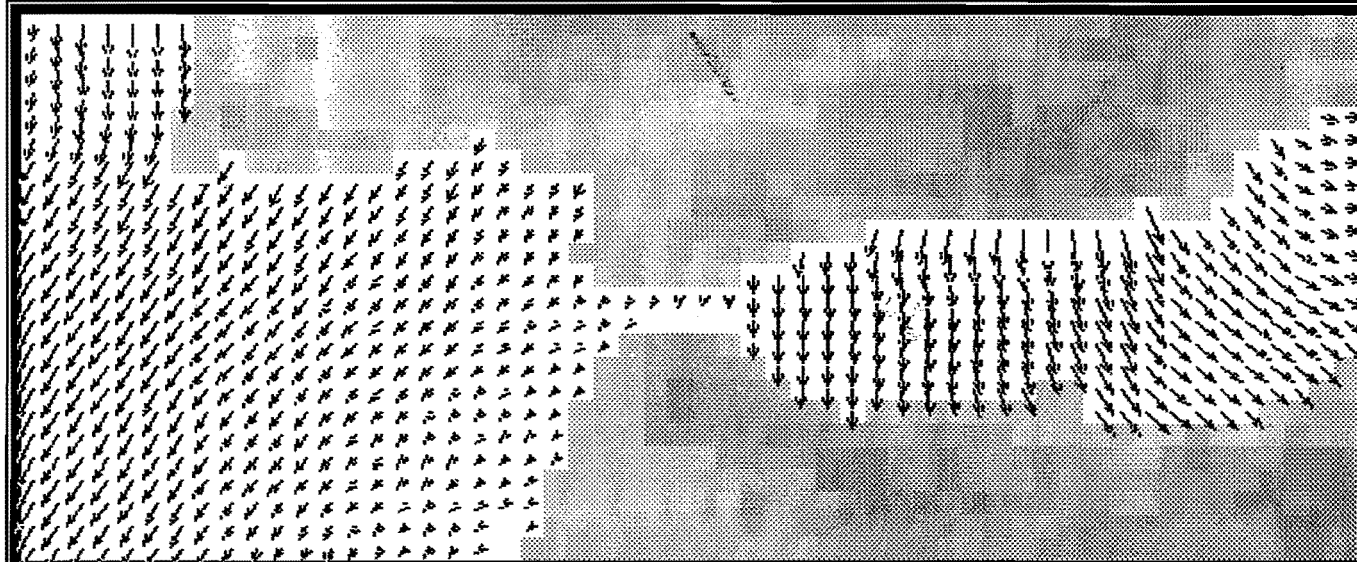
ALCAN = BA

FINISHED, PRESS A KEY
WAIT A MOMENT

UIENTO INICIAL

20 M/S

0



PRESS A KEY TO CONTINUE

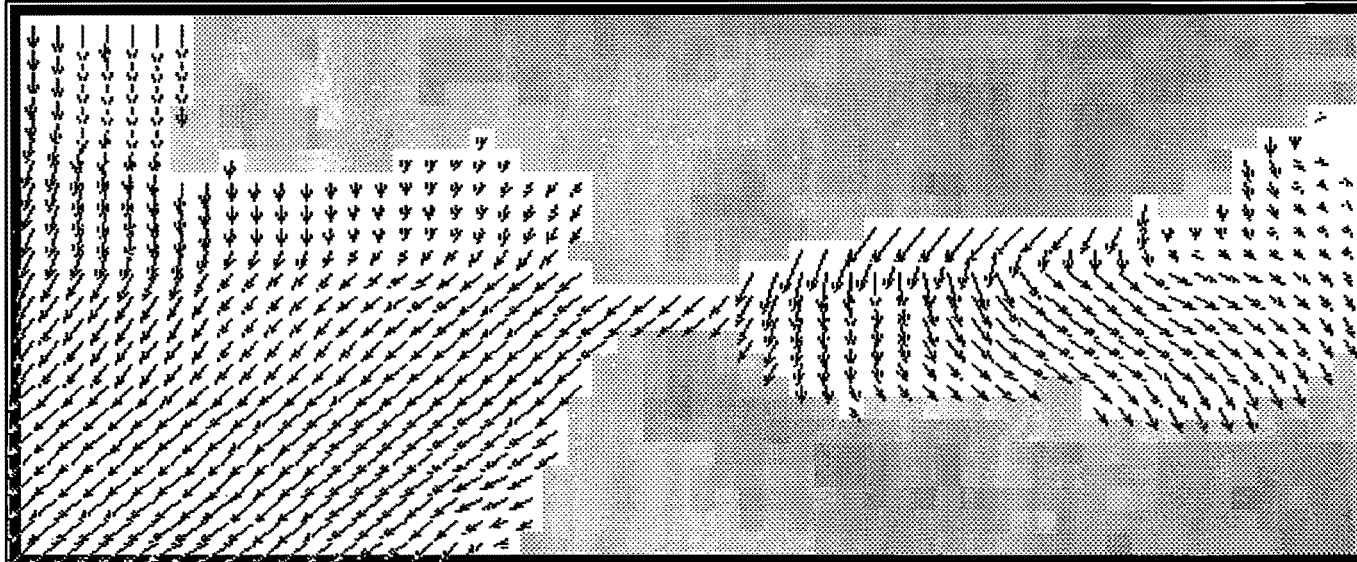
ALCAN = BG

FINISHED, PRESS A KEY
WAIT A MOMENT

UIENTO INICIAL

20 M/S

0



PRESS A KEY TO CONTINUE

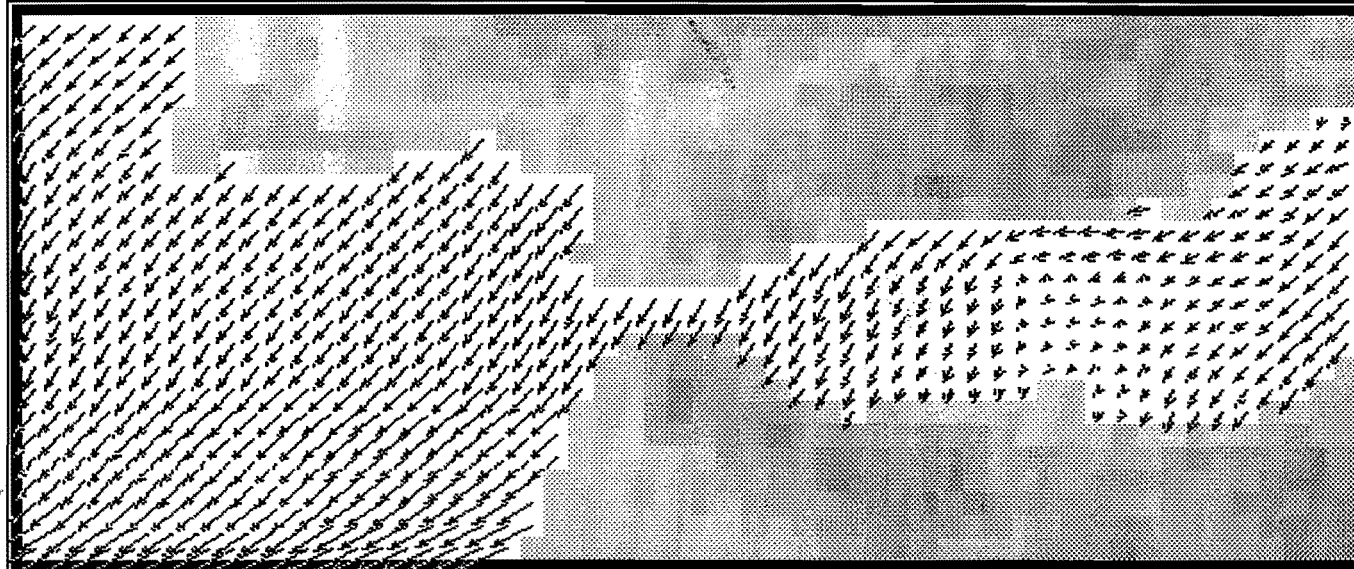
ALCAN = CA

FINISHED, PRESS A KEY
WAIT A MOMENT

UIENTO INICIAL

20 M/S

0



PRESS A KEY TO CONTINUE

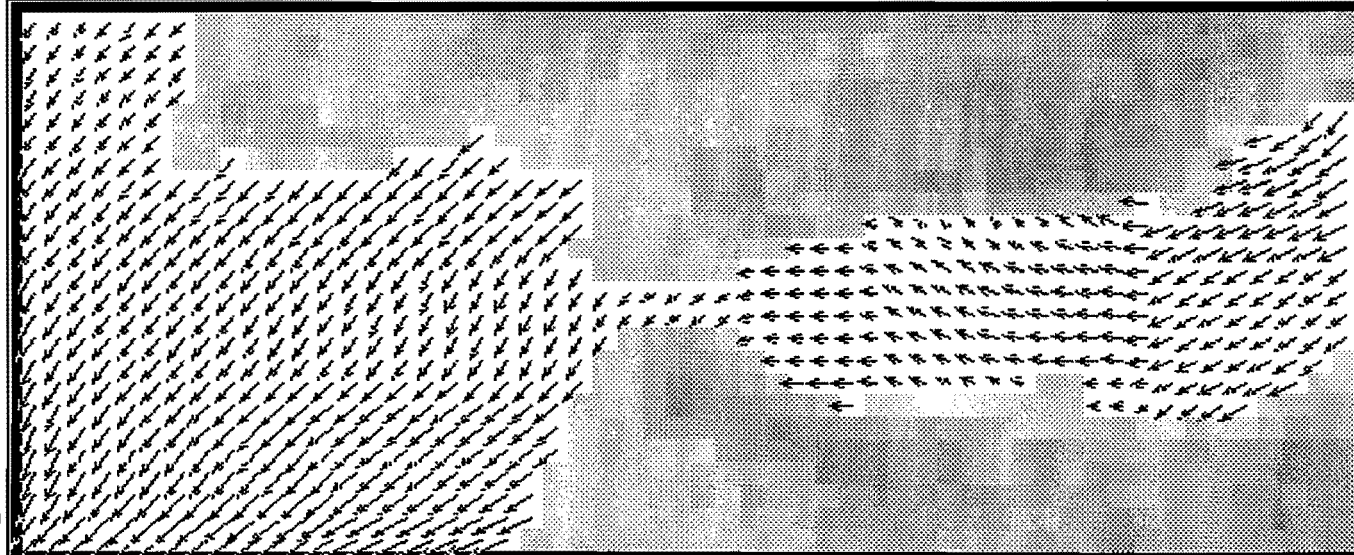
ALCAN = DA

FINISHED, PRESS A KEY
WAIT A MOMENT

UIENTO INICIAL

20 M/S

0



PRESS A KEY TO CONTINUE

ALCAN = EA

2ª Serie

**Efectos:
Rozamiento
Coriolis**

**Variables:
CROZ = 1
CORIOLIS = T, F**

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = AG CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00

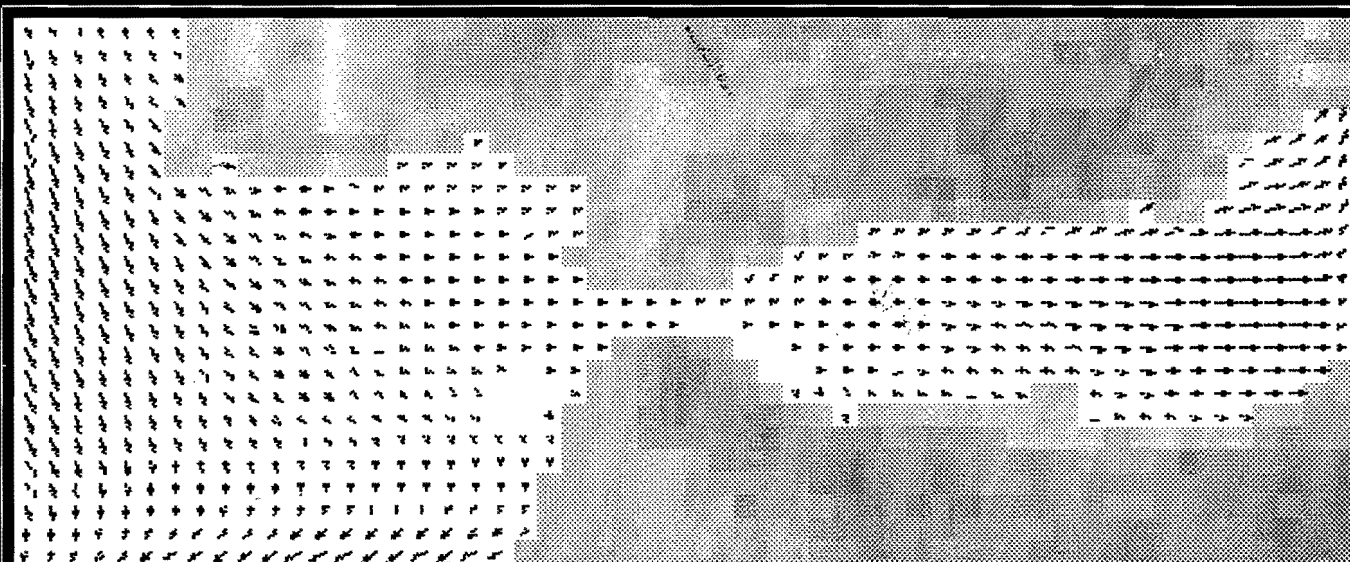
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = AG CORIOL = F UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00

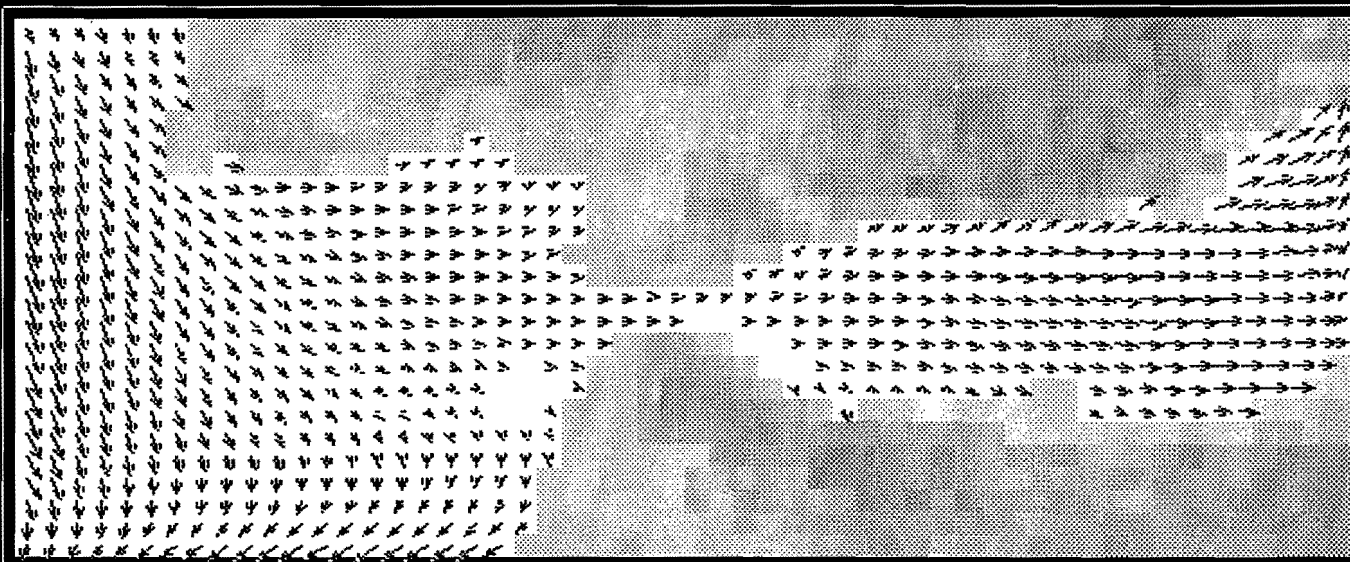
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BA CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00

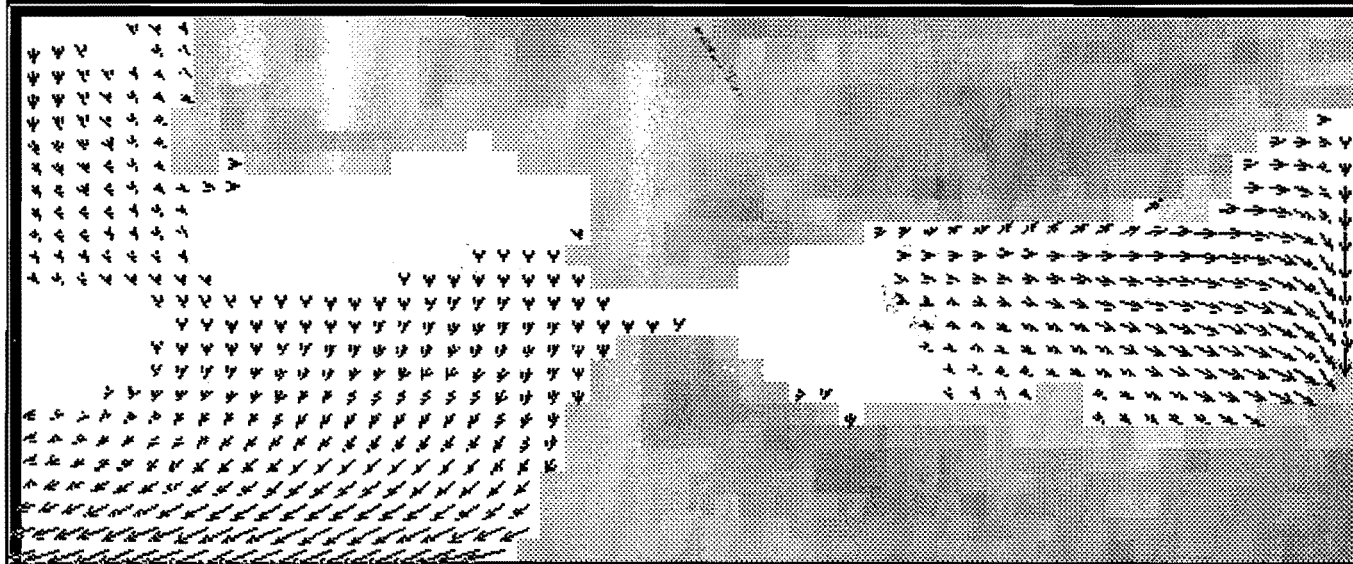
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BA CORIOL = F UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00

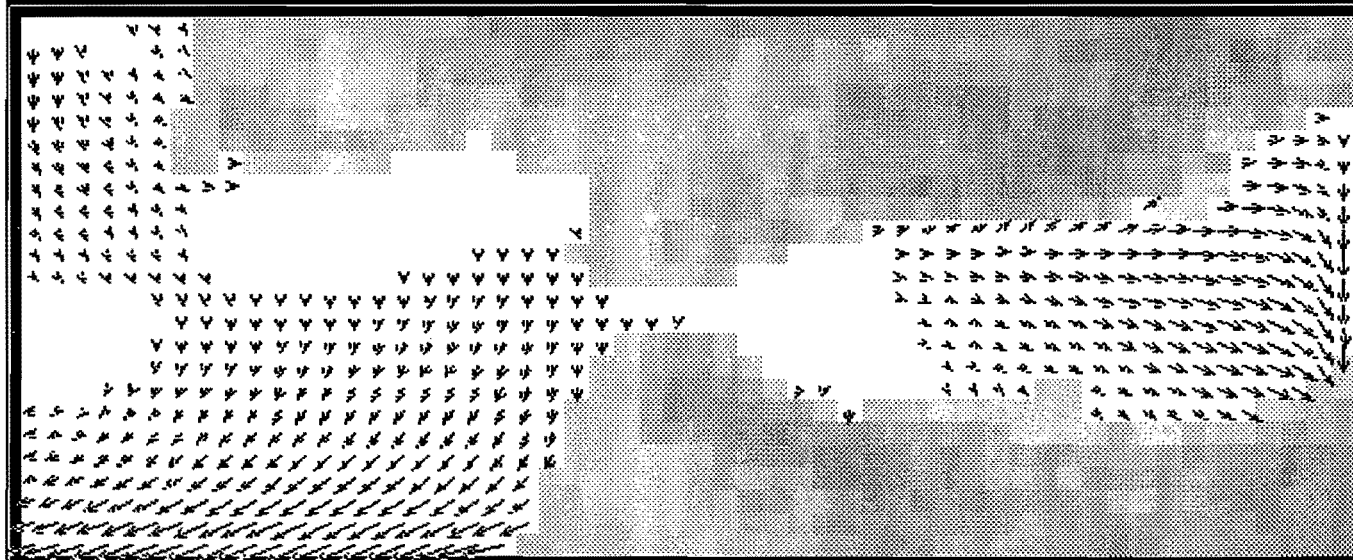
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BG CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00

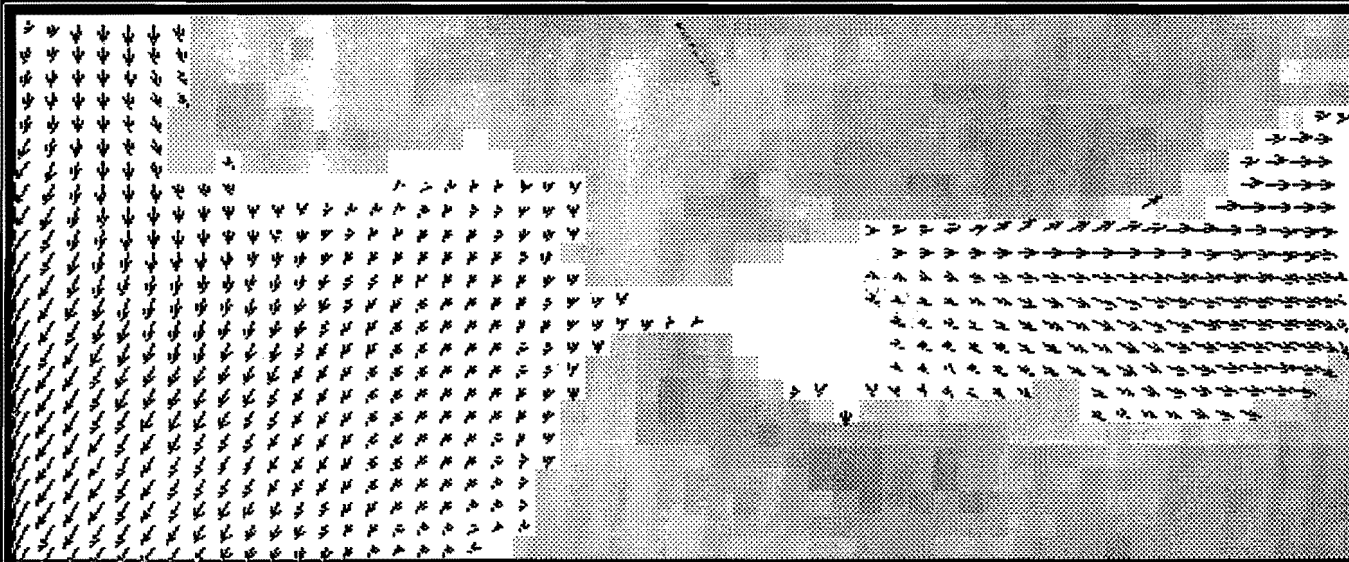
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BG CORIOL = F UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00

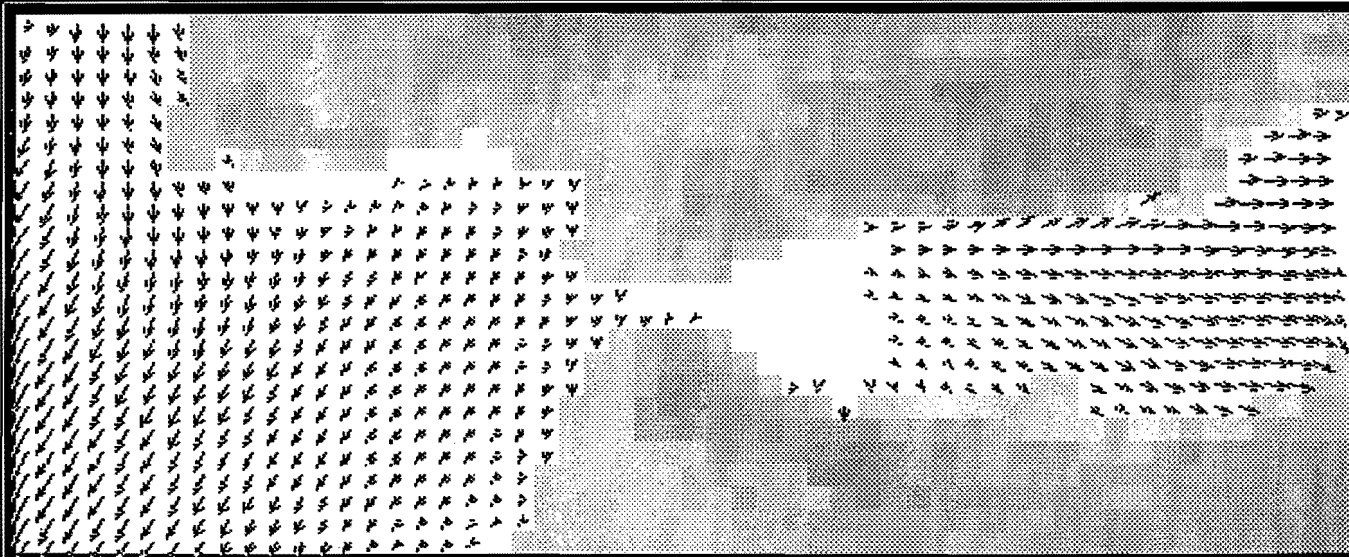
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = CA CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00

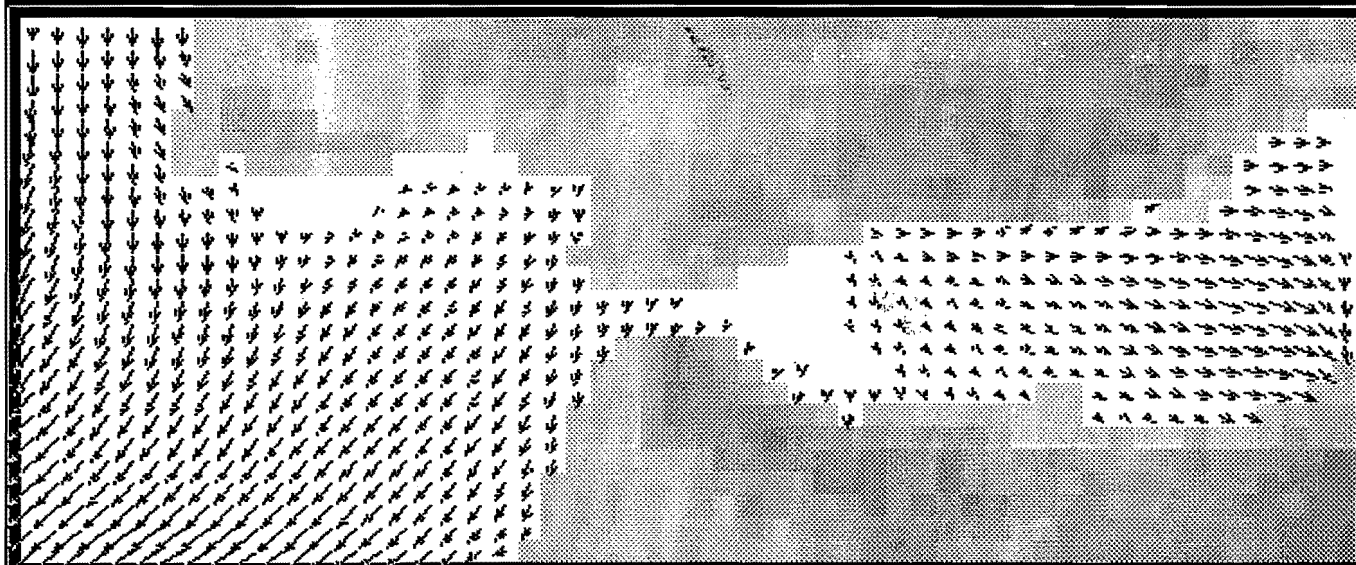
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = CA CORIOL = F UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00

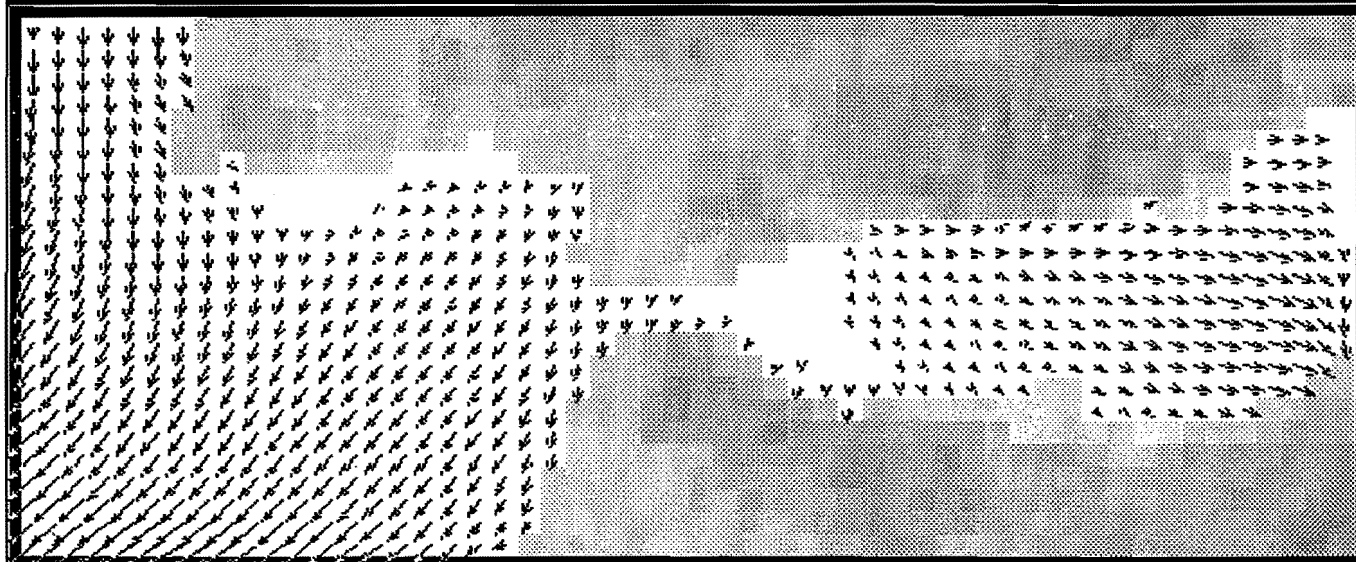
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = DA CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00

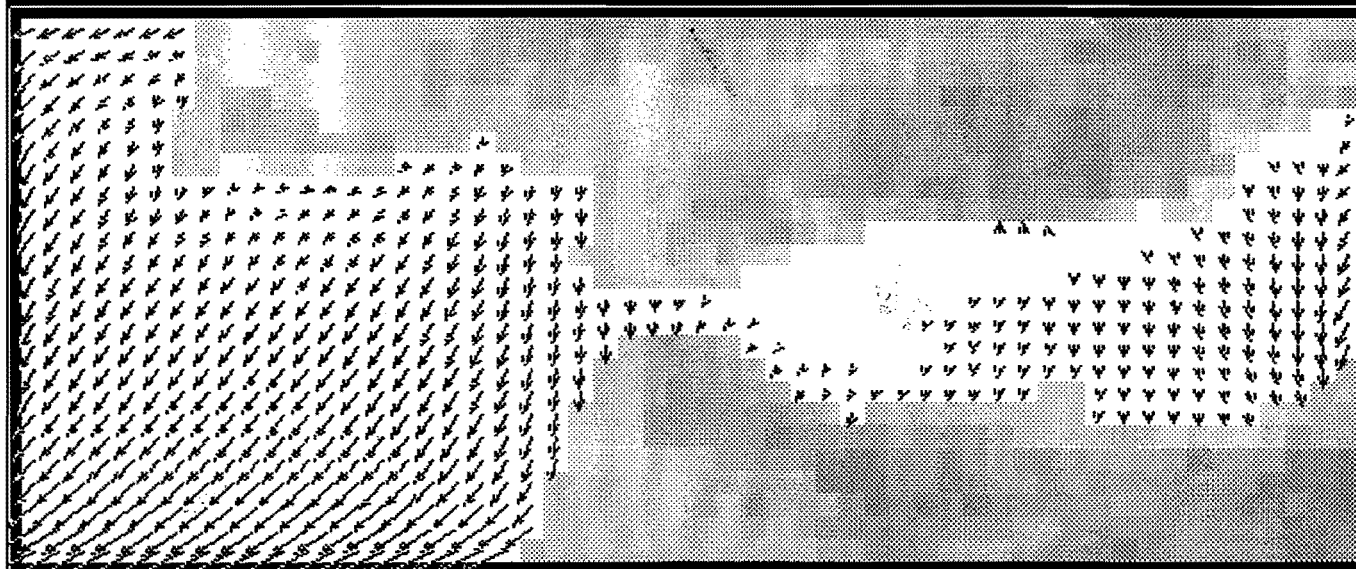
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = DA CORIOL = F UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00

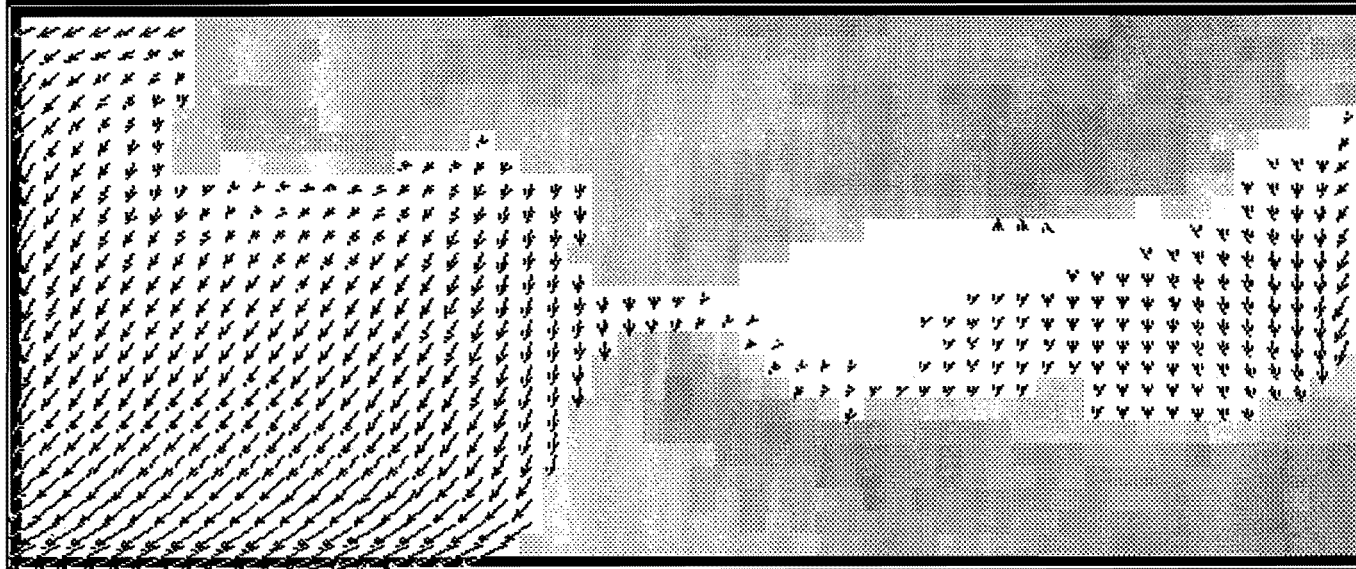
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = EA CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00

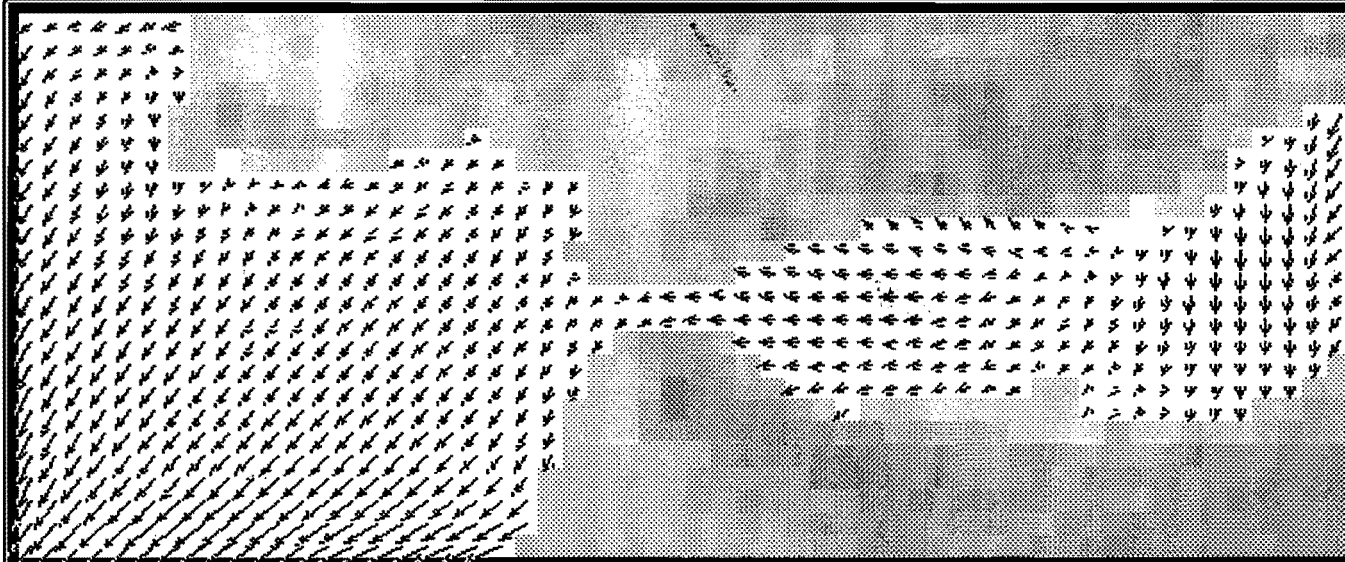
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = EA CORIOL = F UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00

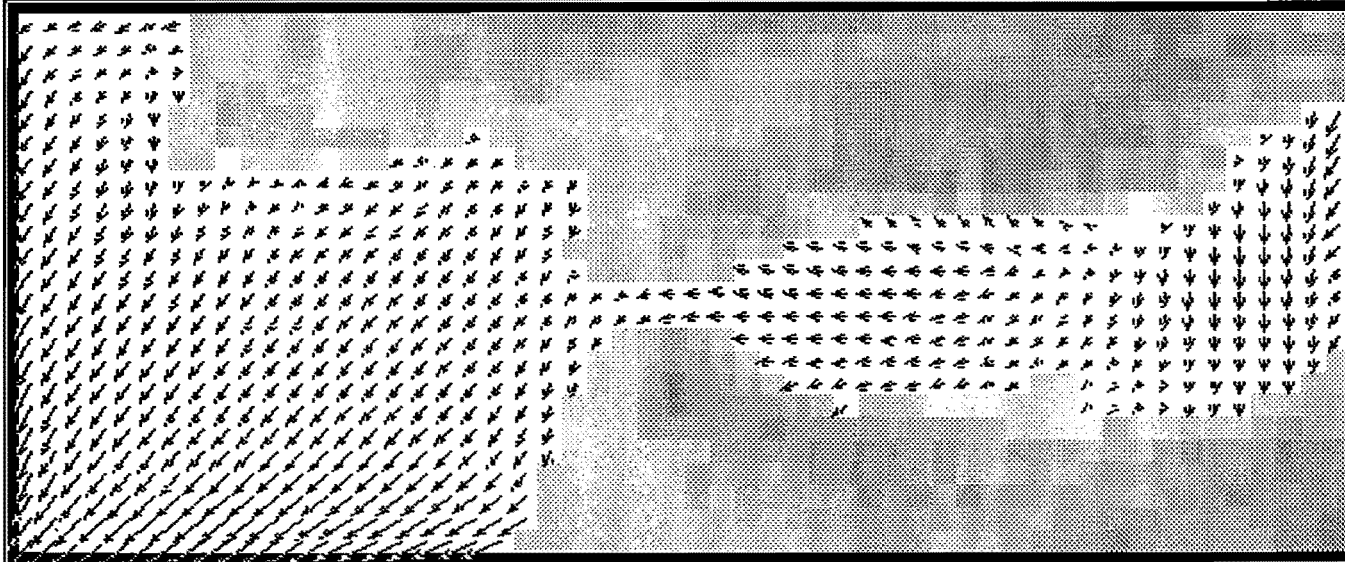
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

3^a Serie

**Efectos:
Rozamiento
Coriolis**

**Variables:
CROZ = 1/2
CORIOLIS = T, F**

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = AG CORIOL = T UGEOST = T TSUP = F

CROZ = 0.50 FACT = 1.00 EPSI = 1.00

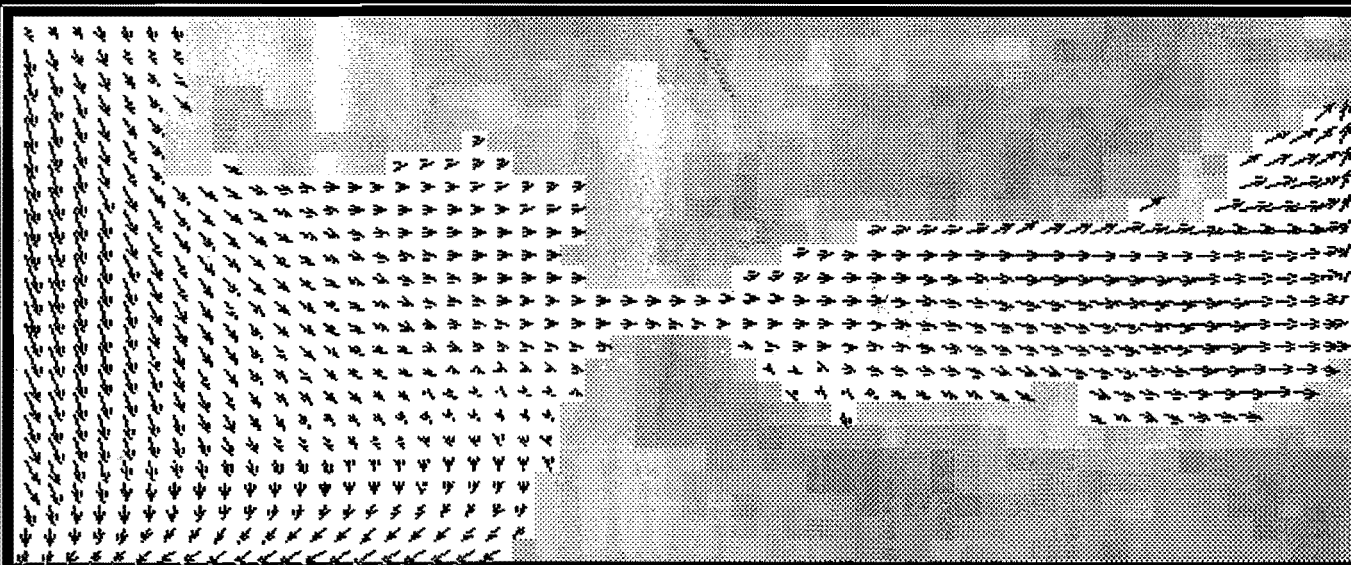
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = AG CORIOL = F UGEOST = T TSUP = F

CROZ = 0.50 FACT = 1.00 EPSI = 1.00

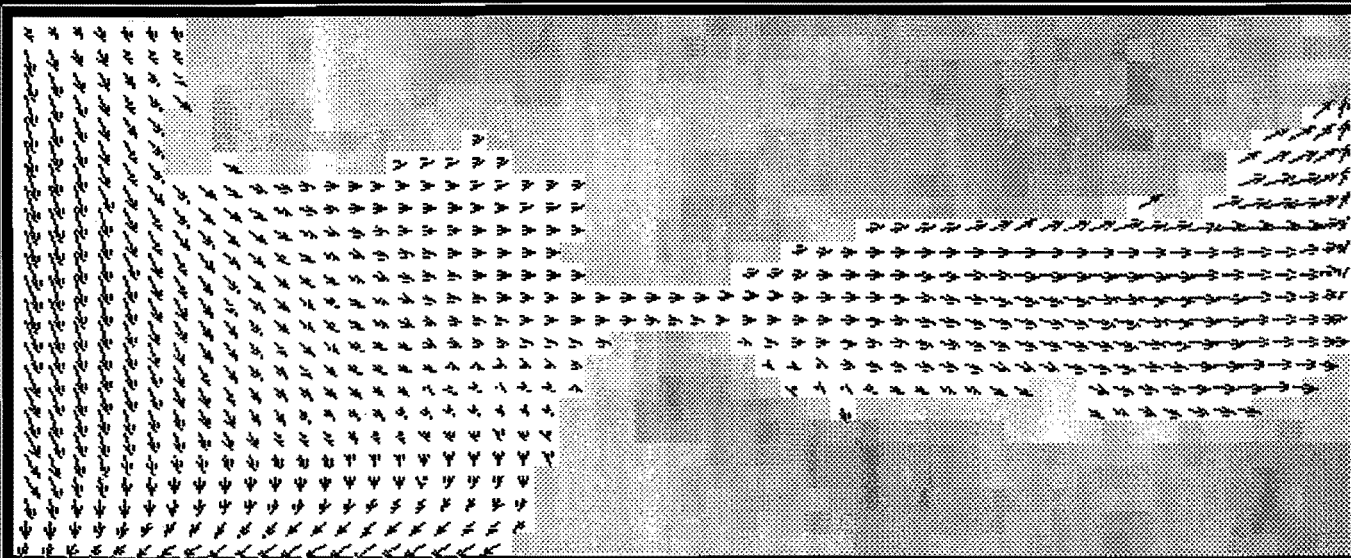
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BA CORIOL = T UGEOST = T TSUP = F

CROZ = 0.50 FACT = 1.00 EPSI = 1.00

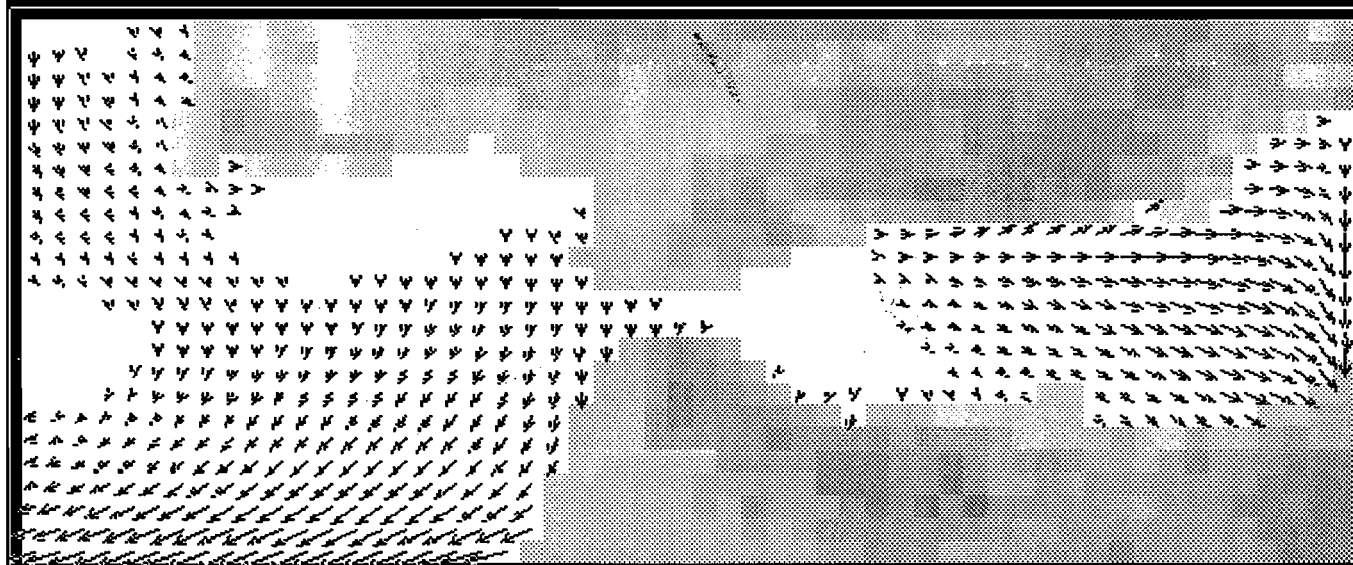
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BA CORIOL = F UGEOST = T TSUP = F

CROZ = 0.50 FACT = 1.00 EPSI = 1.00

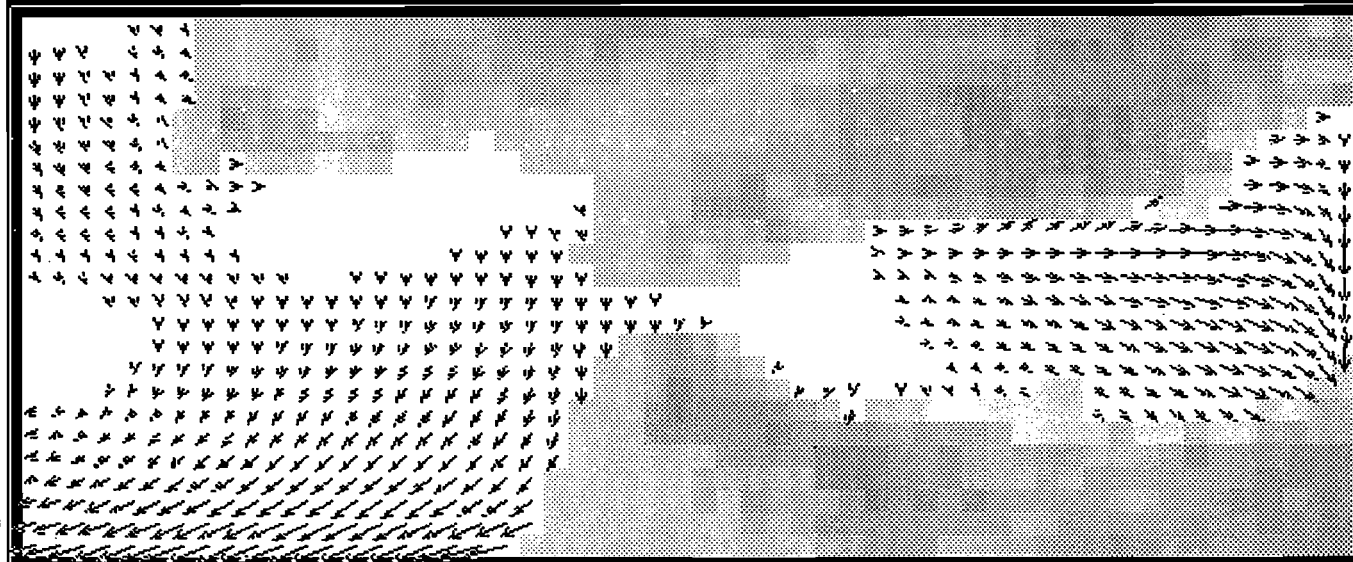
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BG CORIOL = T UGEOST = T TSUP = F

CROZ = 0.50 FACT = 1.00 EPSI = 1.00

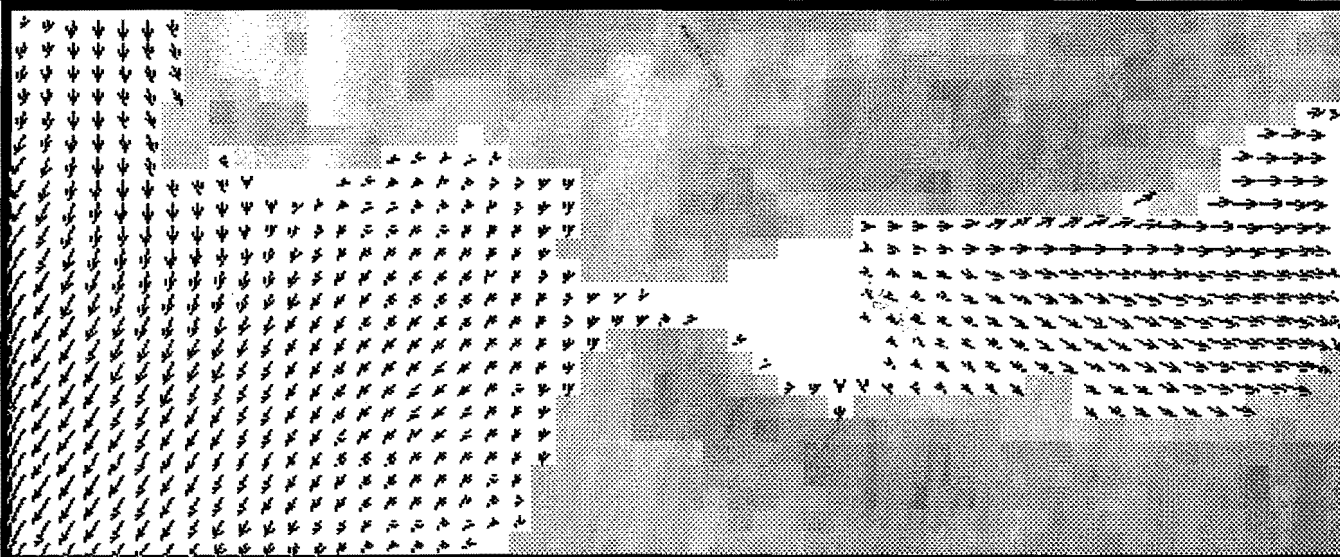
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BG CORIOL = F UGEOST = T TSUP = F

CROZ = 0.50 FACT = 1.00 EPSI = 1.00

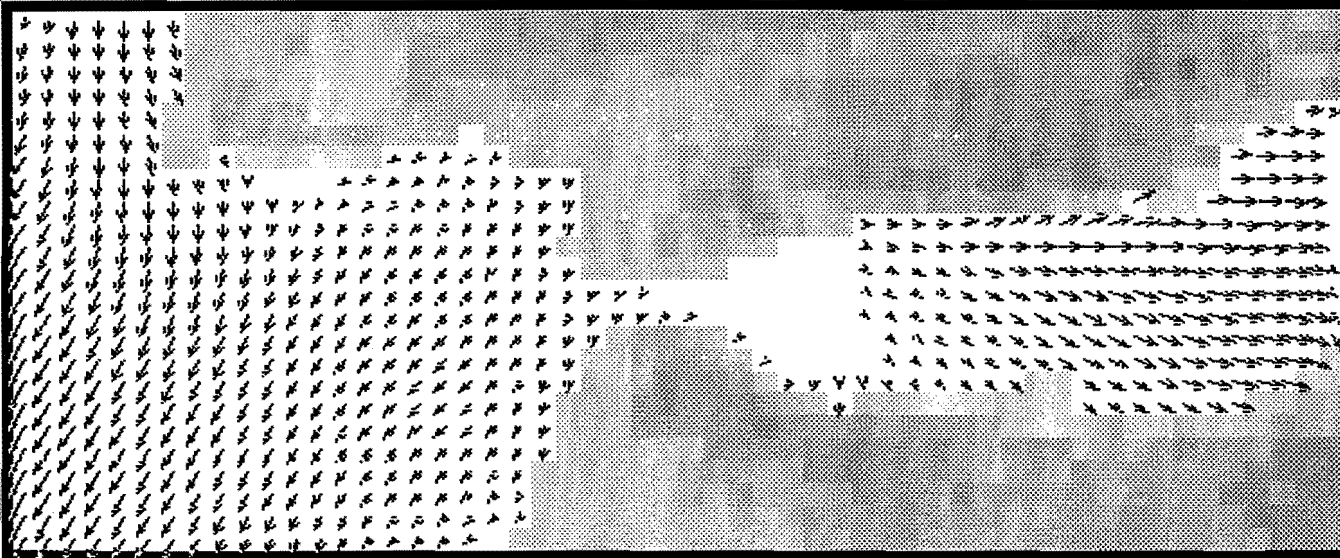
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = CA CORIOL = T UGEOST = T TSUP = F

CROZ = 0.50 FACT = 1.00 EPSI = 1.00

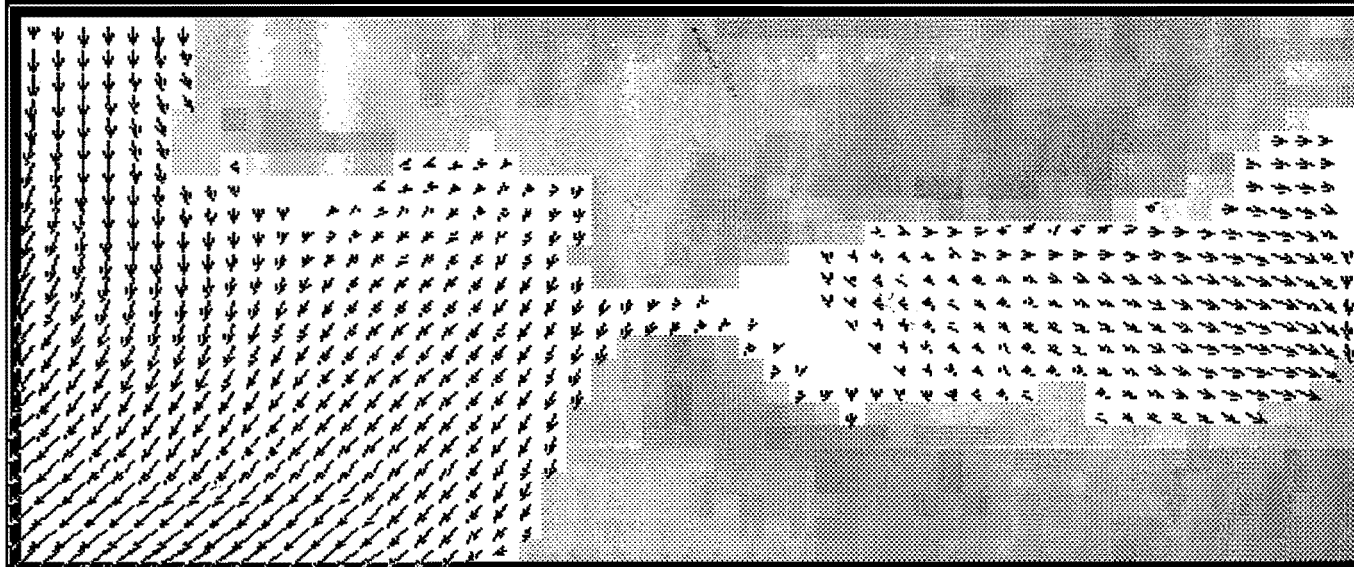
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

WIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = CA CORIOL = F UGEOST = T TSUP = F

CROZ = 0.50 FACT = 1.00 EPSI = 1.00

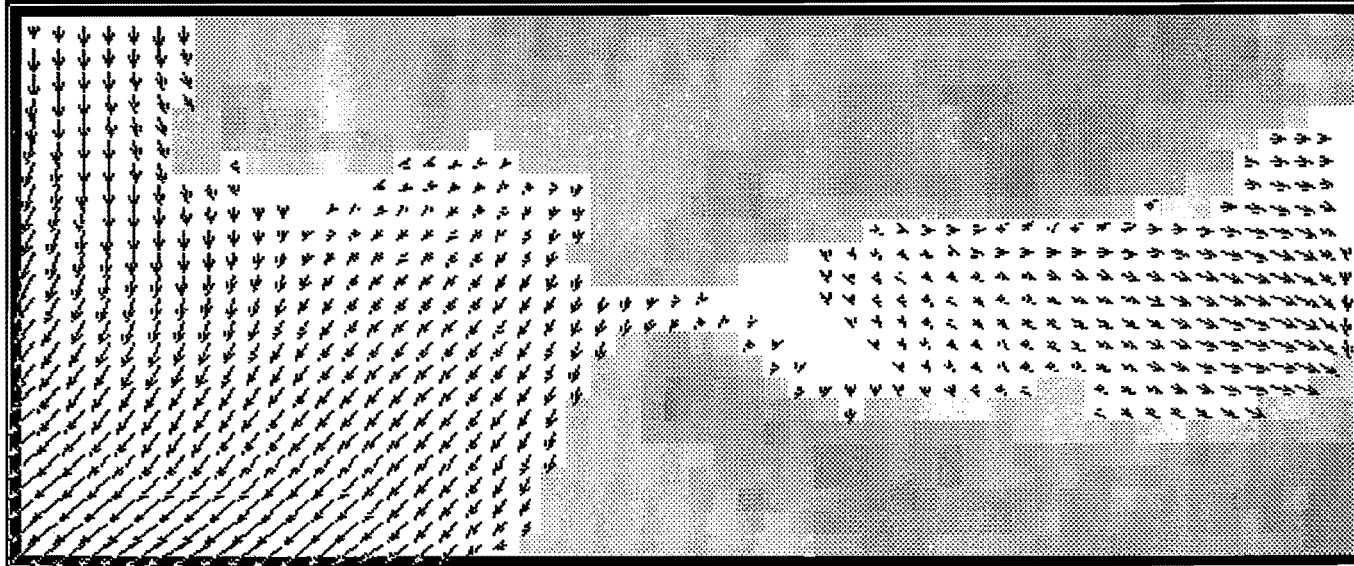
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

WIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = DA CORIOL = T UGEOST = T TSUP = F

CROZ = 0.50 FACT = 1.00 EPSI = 1.00

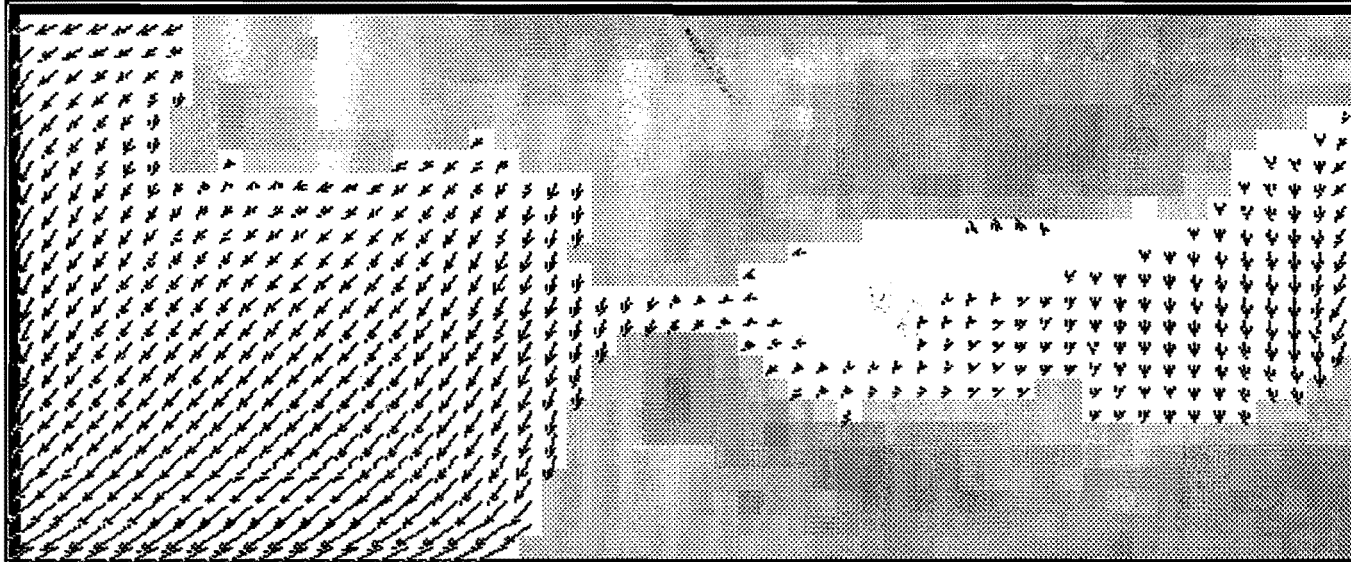
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = DA CORIOL = F UGEOST = T TSUP = F

CROZ = 0.50 FACT = 1.00 EPSI = 1.00

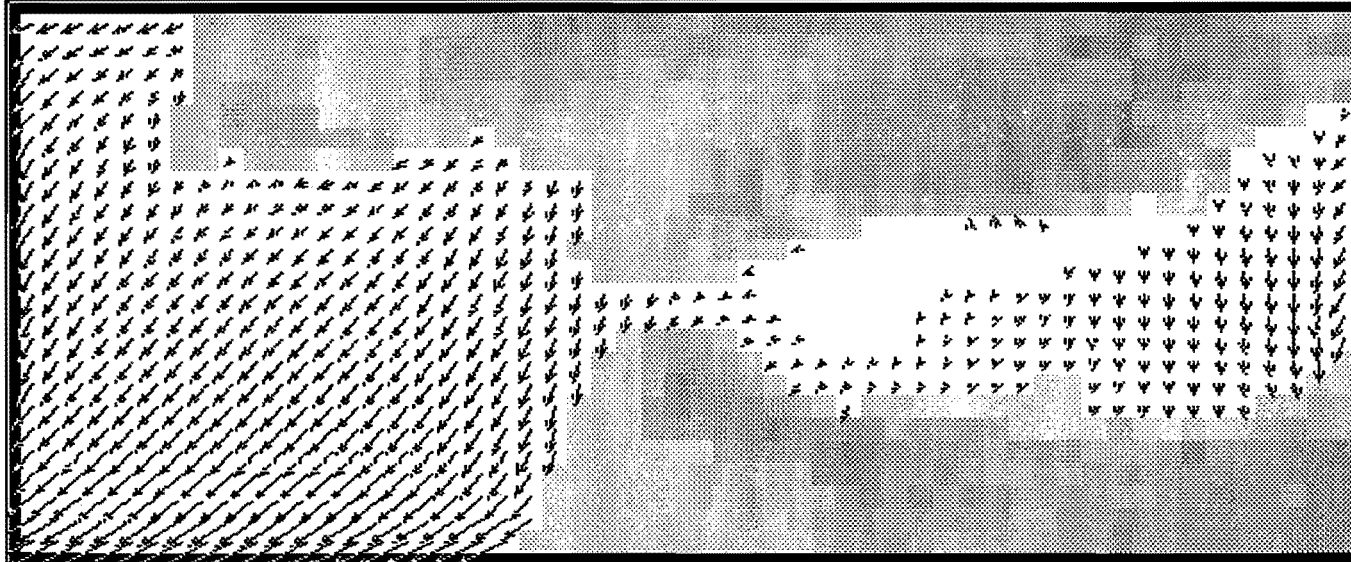
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = EA CORIOL = T UGEOST = T TSUP = F

CROZ = 0.50 FACT = 1.00 EPSI = 1.00

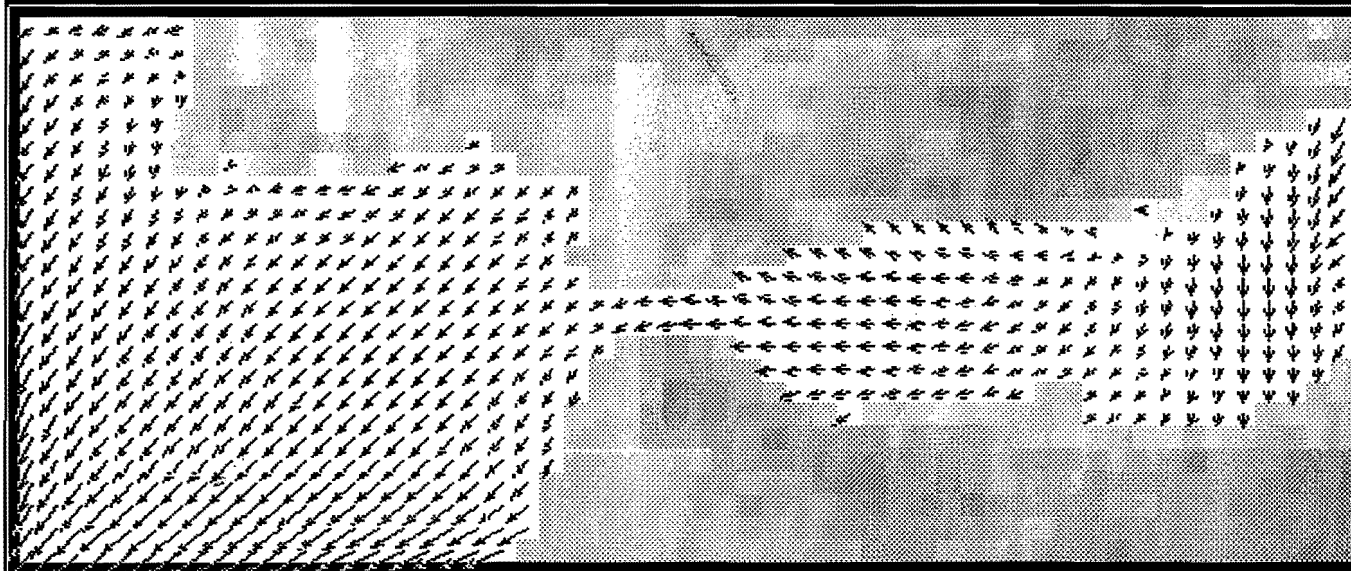
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

WIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = EA CORIOL = F UGEOST = T TSUP = F

CROZ = 0.50 FACT = 1.00 EPSI = 1.00

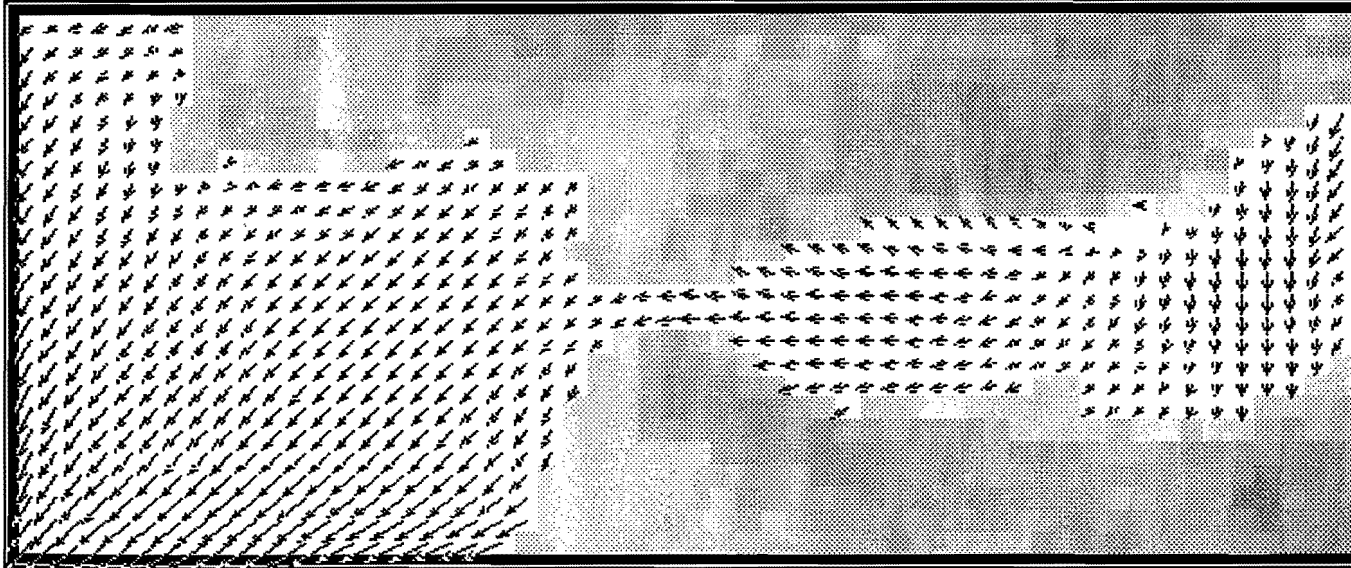
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

WIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

4ª Serie

**Efectos:
Rozamiento
Coriolis**

**Variables:
CROZ = 1/5
CORIOLIS = T, F**

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = AG CORIOL = T UGEOST = T TSUP = F

CROZ = 0.20 FACT = 1.00 EPSI = 1.00

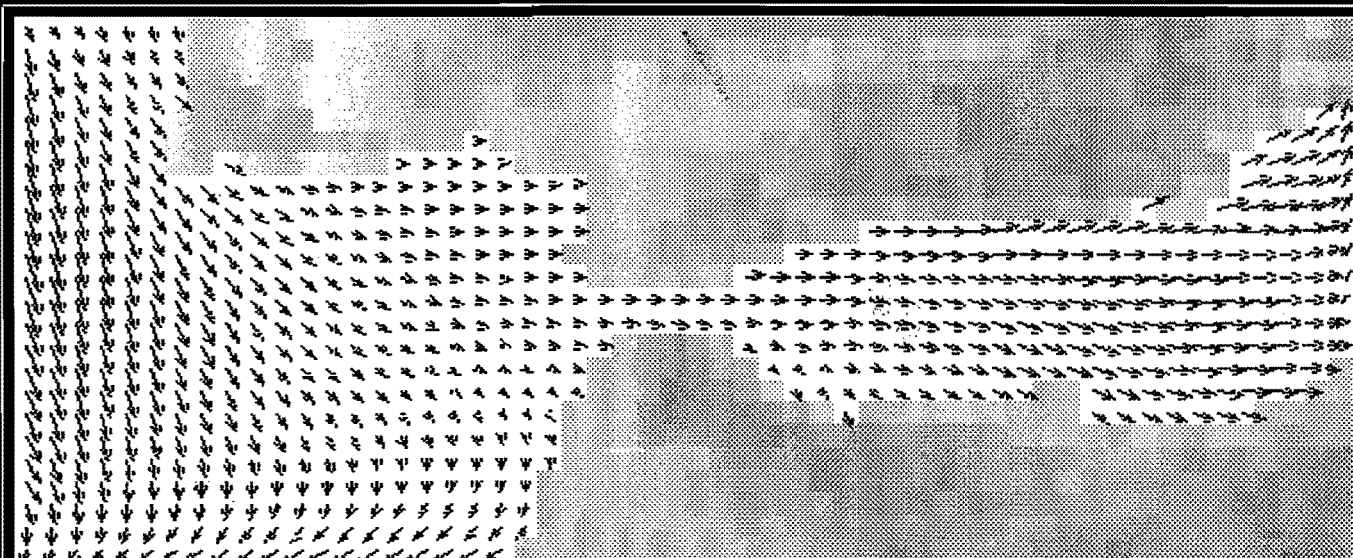
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = AG CORIOL = F UGEOST = T TSUP = F

CROZ = 0.20 FACT = 1.00 EPSI = 1.00

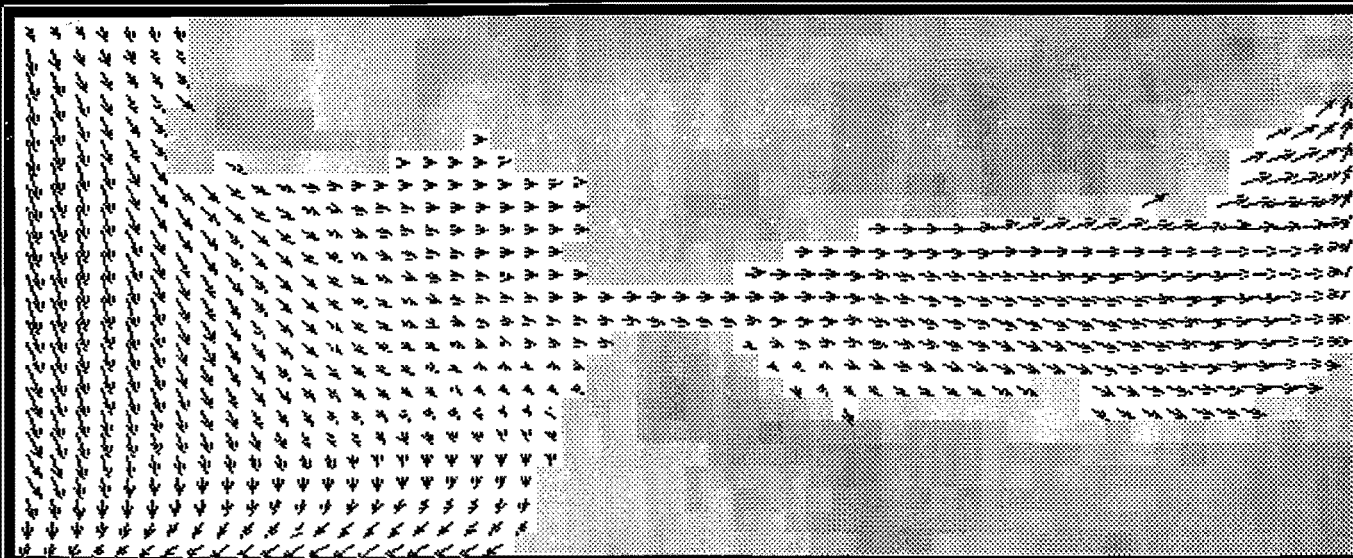
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BA CORIOL = T UGEOST = T TSUP = F

CROZ = 0.20 FACT = 1.00 EPSI = 1.00

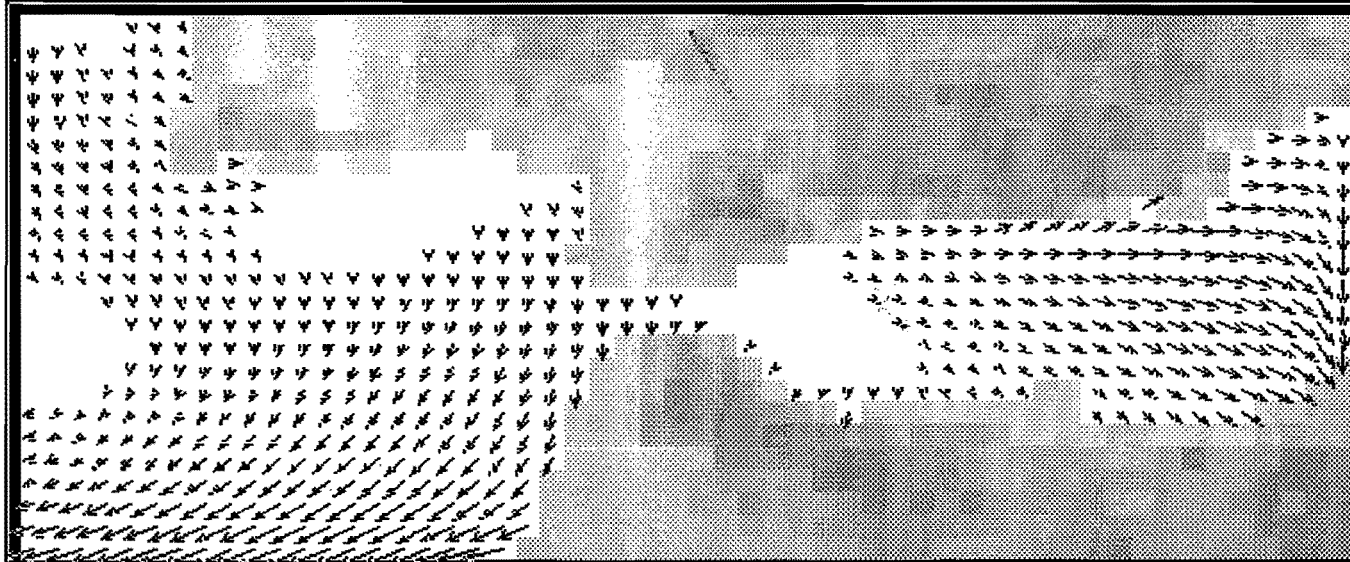
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

WINDO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BA CORIOL = F UGEOST = T TSUP = F

CROZ = 0.20 FACT = 1.00 EPSI = 1.00

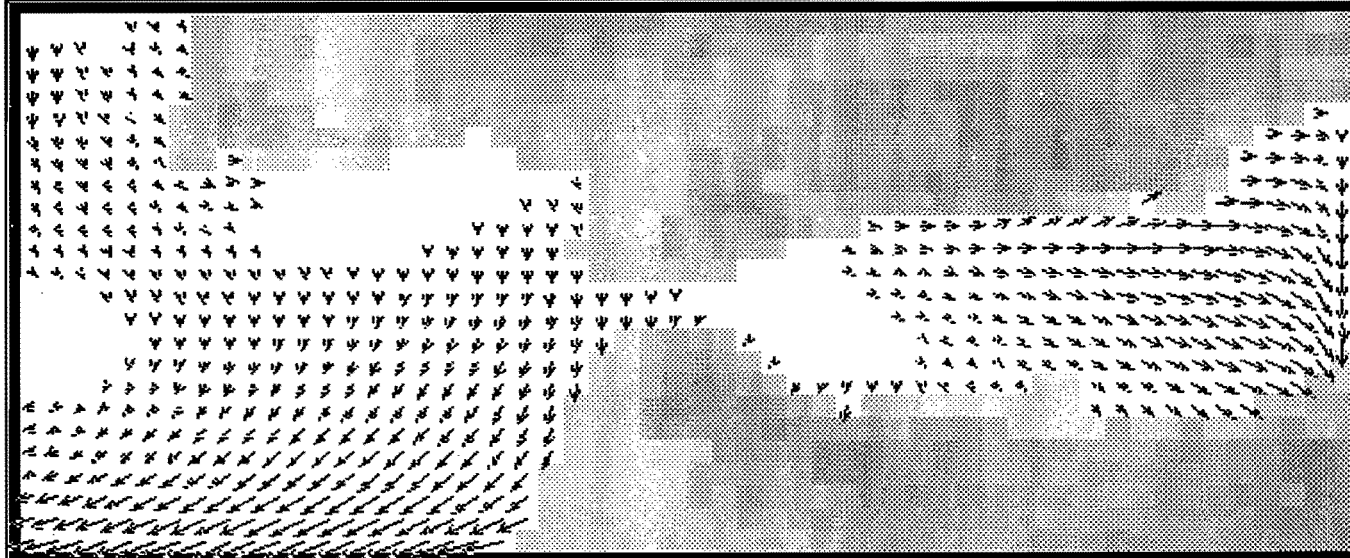
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

WINDO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BG CORIOL = T UGEOST = T TSUP = F

CROZ = 0.20 FACT = 1.00 EPSI = 1.00

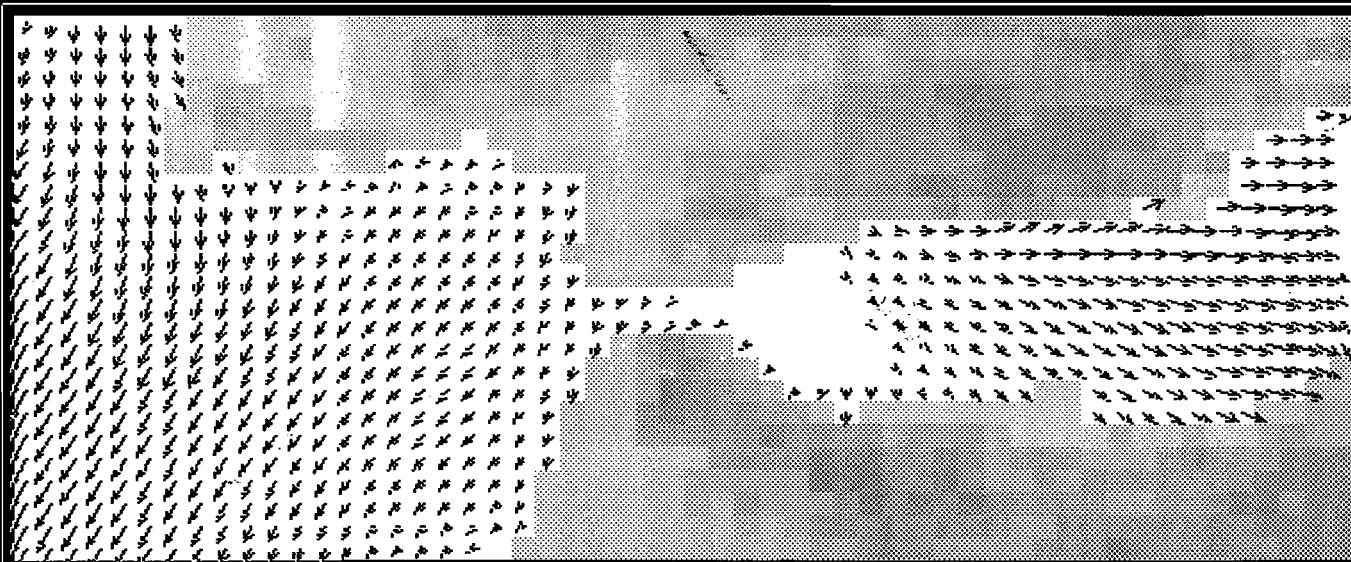
ORDMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

WIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BG CORIOL = F UGEOST = T TSUP = F

CROZ = 0.20 FACT = 1.00 EPSI = 1.00

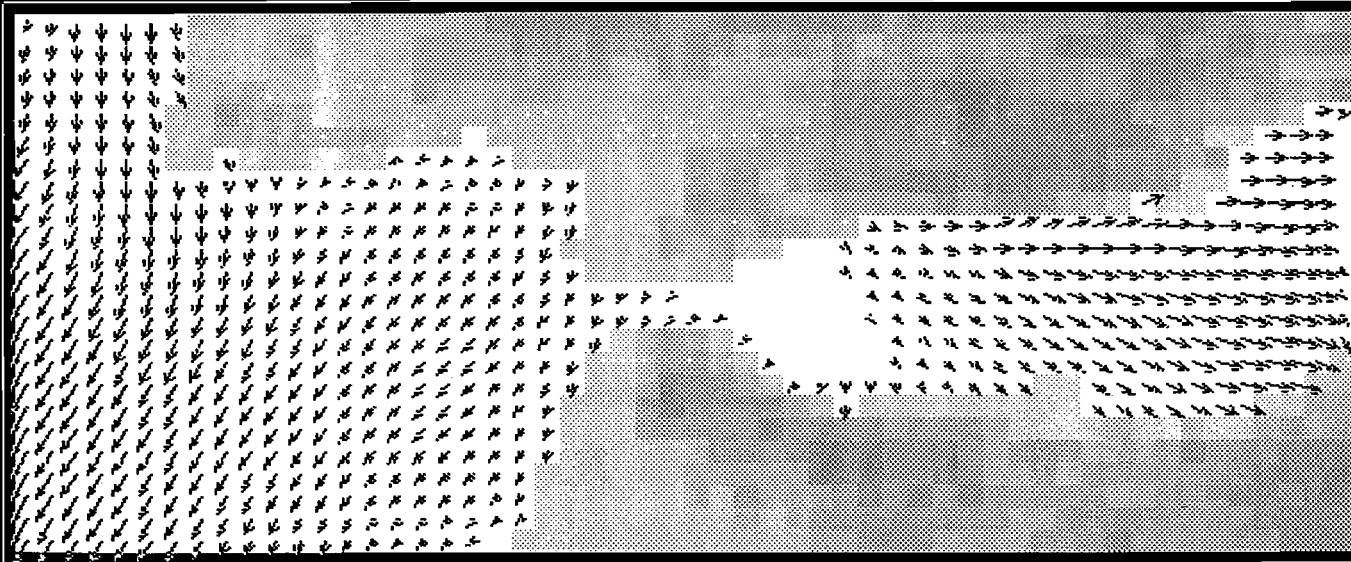
ORDMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

WIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = CA CORIOL = T UGEOST = T TSUP = F

CROZ = 0.20 FACT = 1.00 EPSI = 1.00

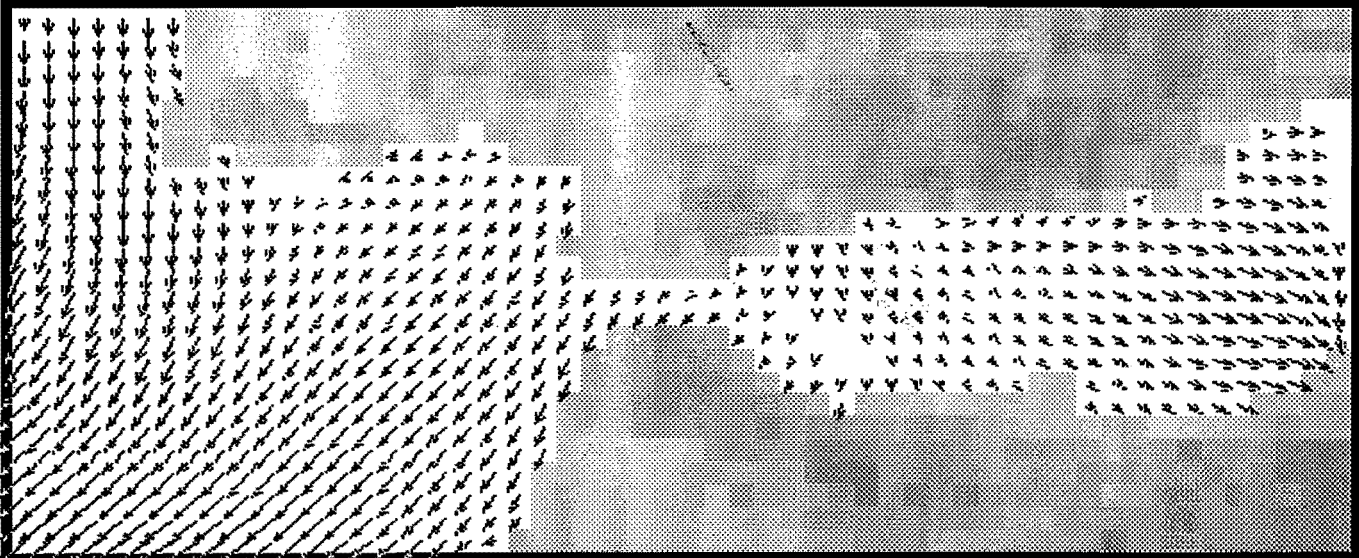
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = CA CORIOL = F UGEOST = T TSUP = F

CROZ = 0.20 FACT = 1.00 EPSI = 1.00

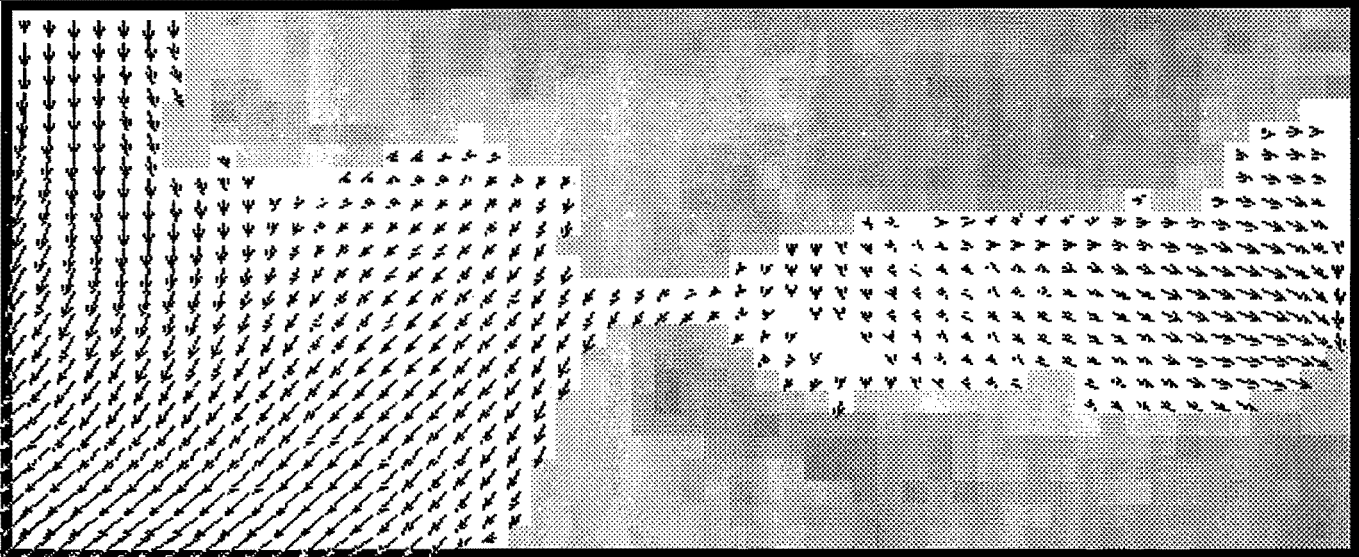
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = DA CORIOL = T UGEOST = T TSUP = F

CROZ = 0.20 FACT = 1.00 EPSI = 1.00

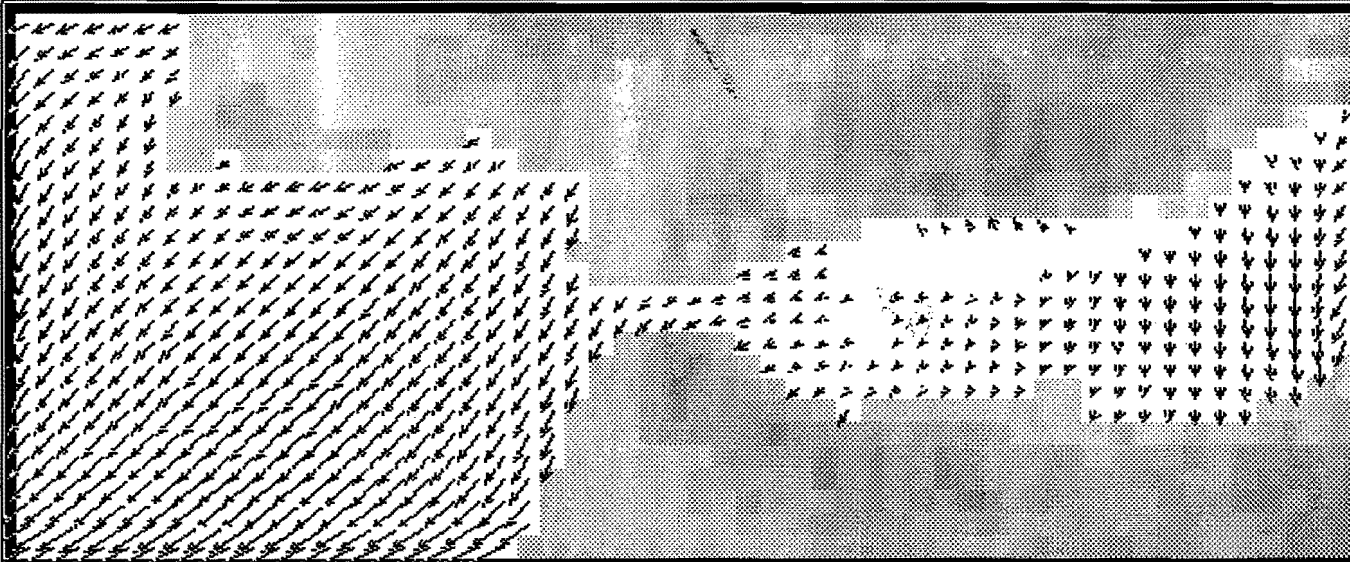
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

UIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = DA CORIOL = F UGEOST = T TSUP = F

CROZ = 0.20 FACT = 1.00 EPSI = 1.00

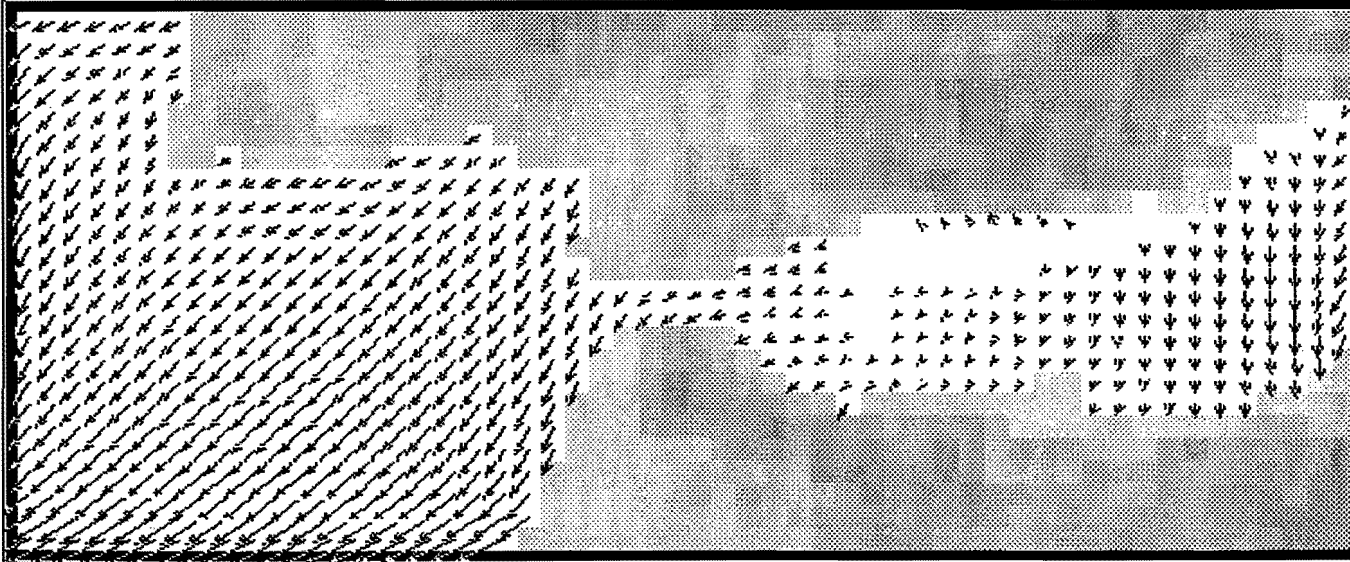
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

UIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = EA CORIOL = T UGEOST = T TSUP = F

CROZ = 0.20 FACT = 1.00 EPSI = 1.00

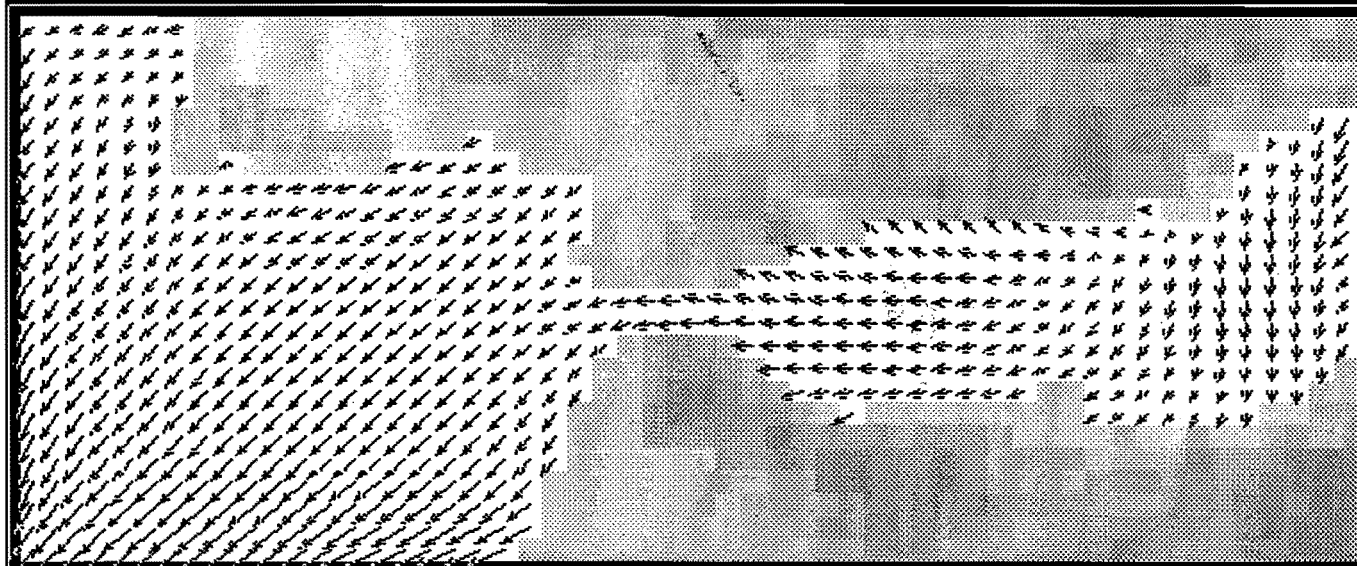
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = EA CORIOL = F UGEOST = T TSUP = F

CROZ = 0.20 FACT = 1.00 EPSI = 1.00

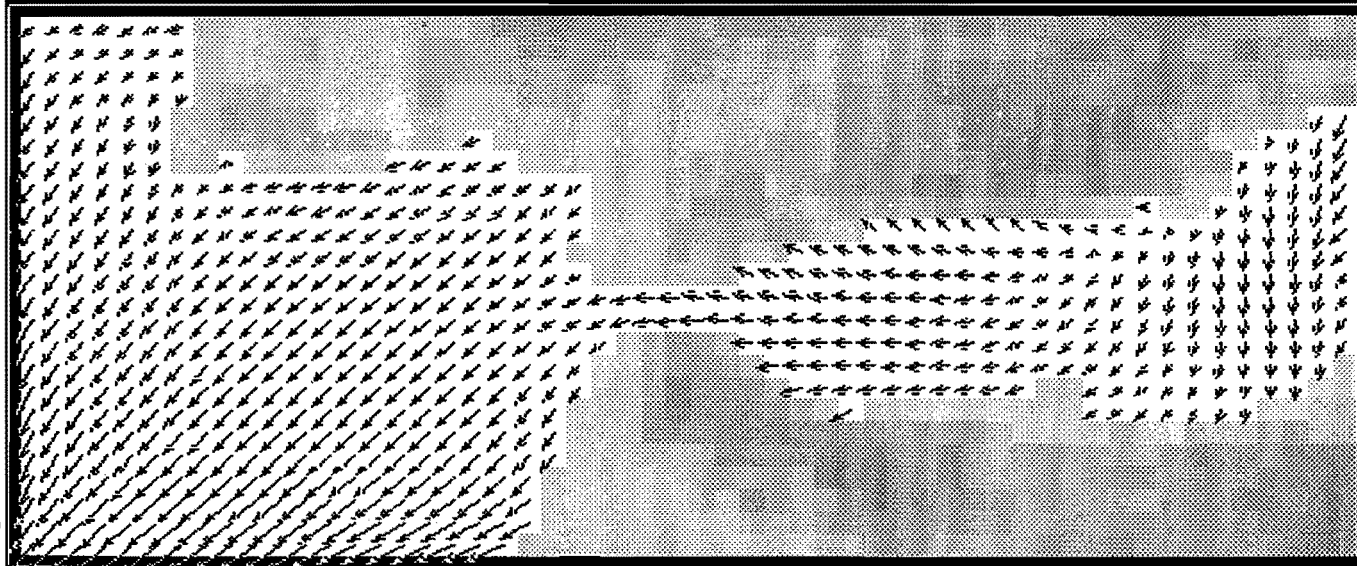
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

5ª Serie

**Efectos:
Gradiente vertical
de temperatura**

**Variables:
GAMMA = 0.004
GAMMA = 0.008**

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = AG CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00 GAMMA = 0.0040

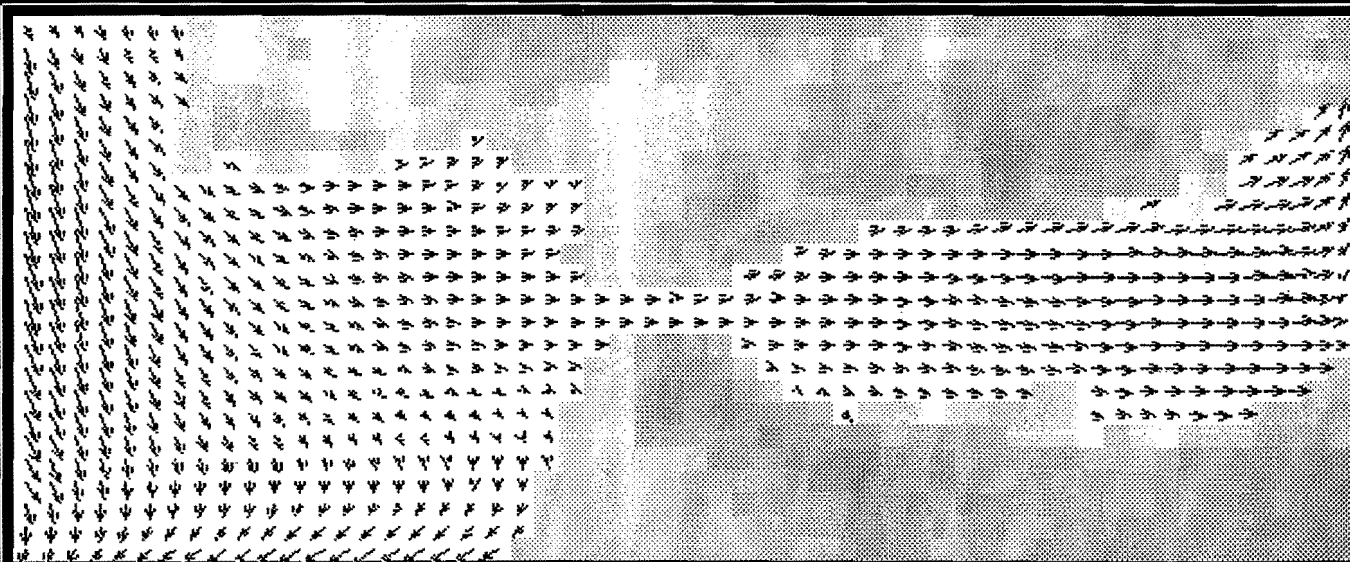
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

WIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = AG CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00 GAMMA = 0.0080

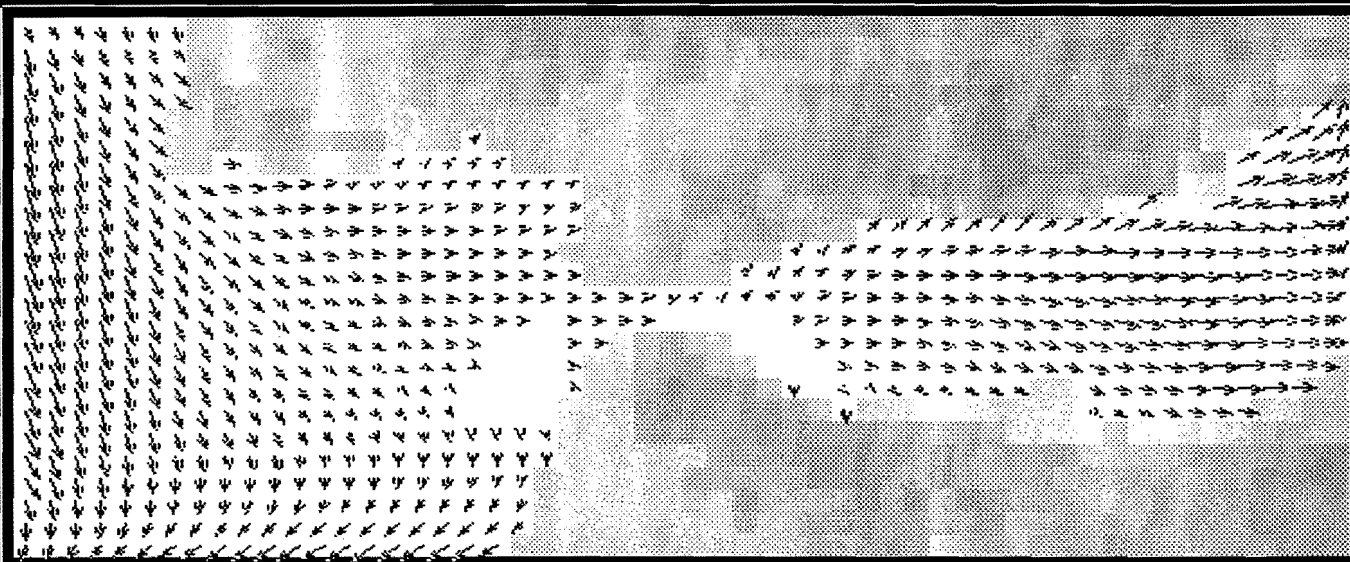
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.0014 CDLAND = 0.0200 ACSEA = 0.0400 ACLAND = 1.0000

WIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BA CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00 GAMMA = 0.0040

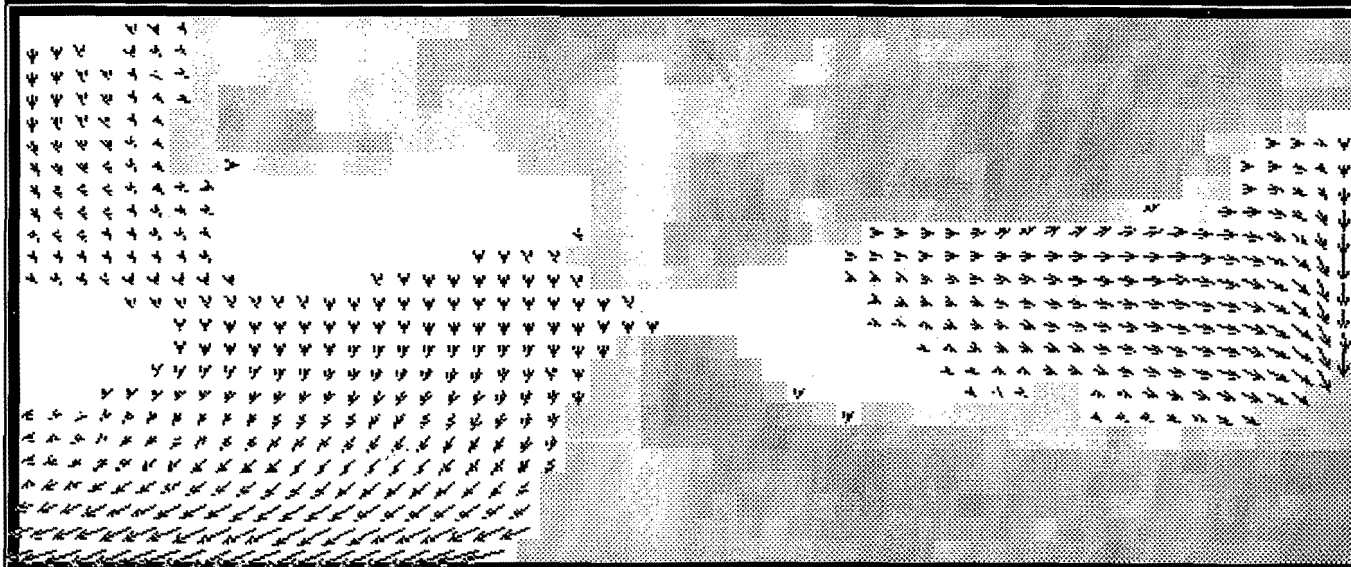
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BA CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00 GAMMA = 0.0080

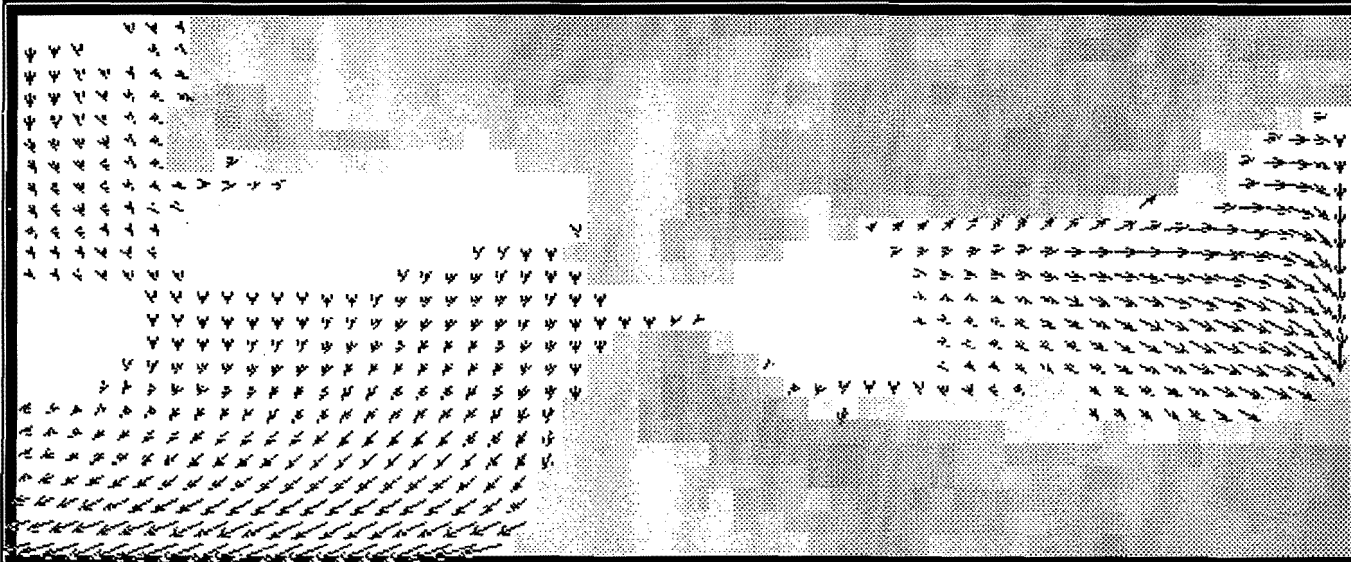
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.0014 CDLAND = 0.0200 ACSEA = 0.0400 ACLAND = 1.0000

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BG CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00 GAMMA = 0.0040

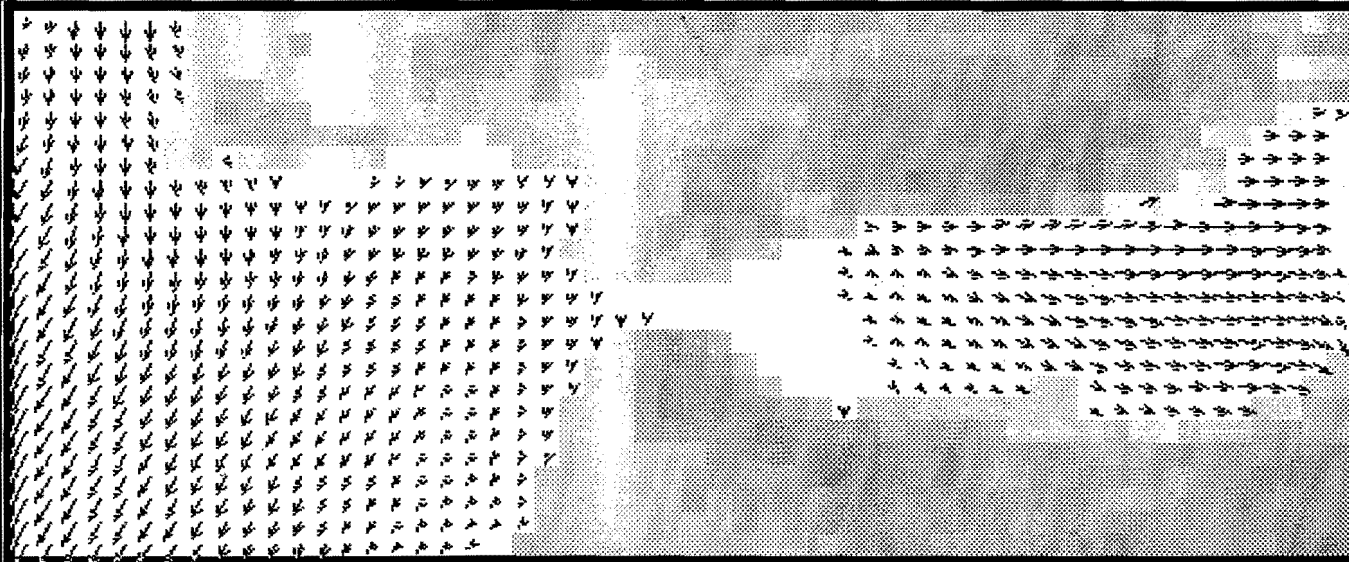
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = BG CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00 GAMMA = 0.0080

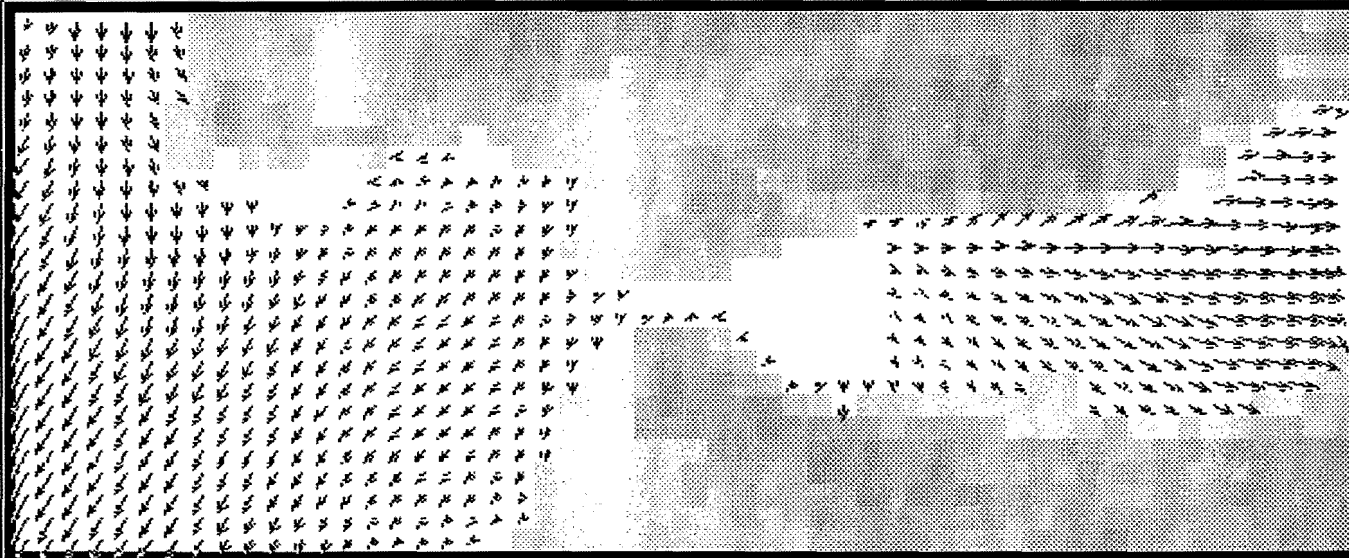
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.0014 CDLAND = 0.0200 ACSEA = 0.0400 ACLAND = 1.0000

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = CA CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00 GAMMA = 0.0040

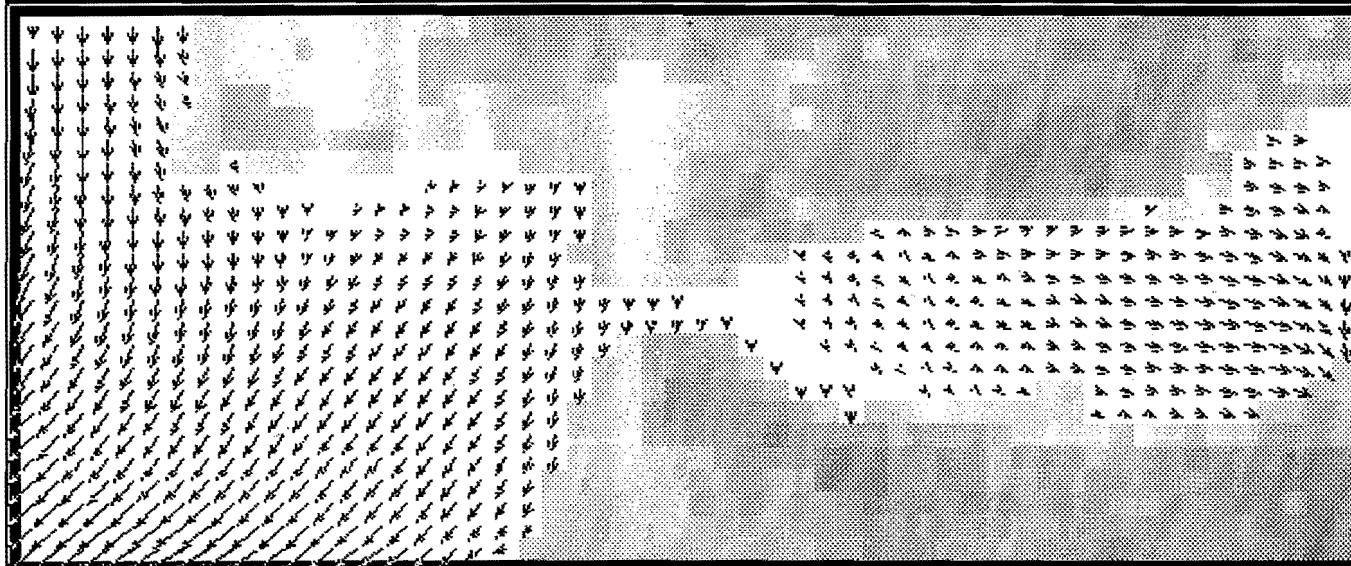
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = CA CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00 GAMMA = 0.0080

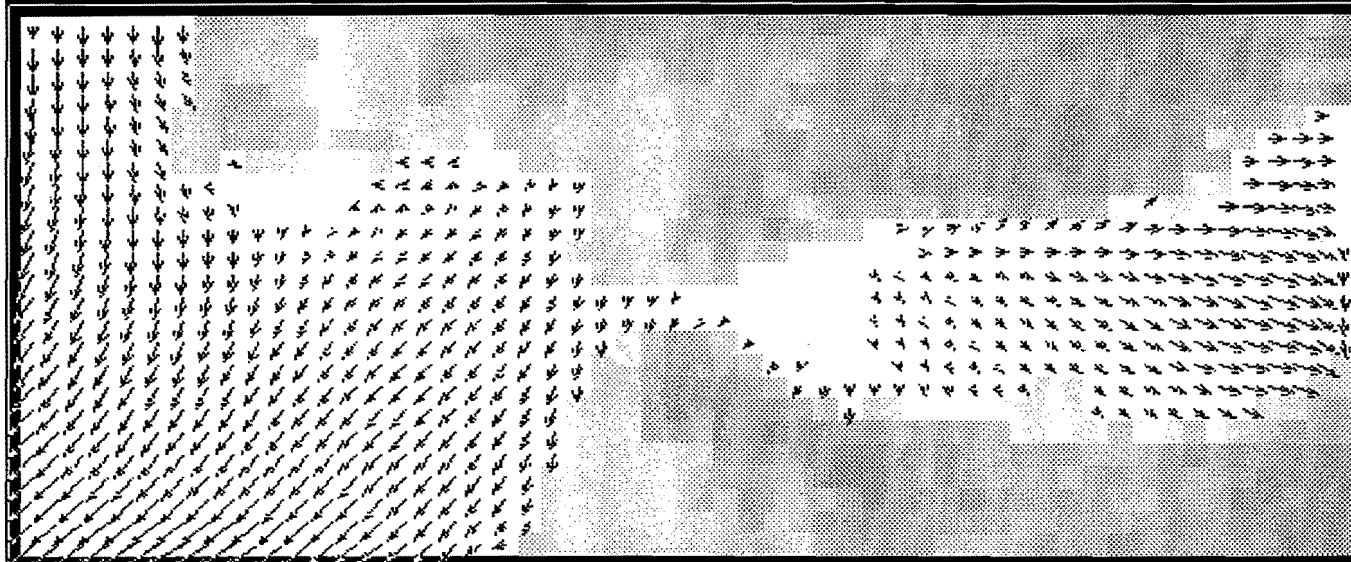
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.0014 CDLAND = 0.0200 ACSEA = 0.0400 ACLAND = 1.0000

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = DA CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00 GAMMA = 0.0040

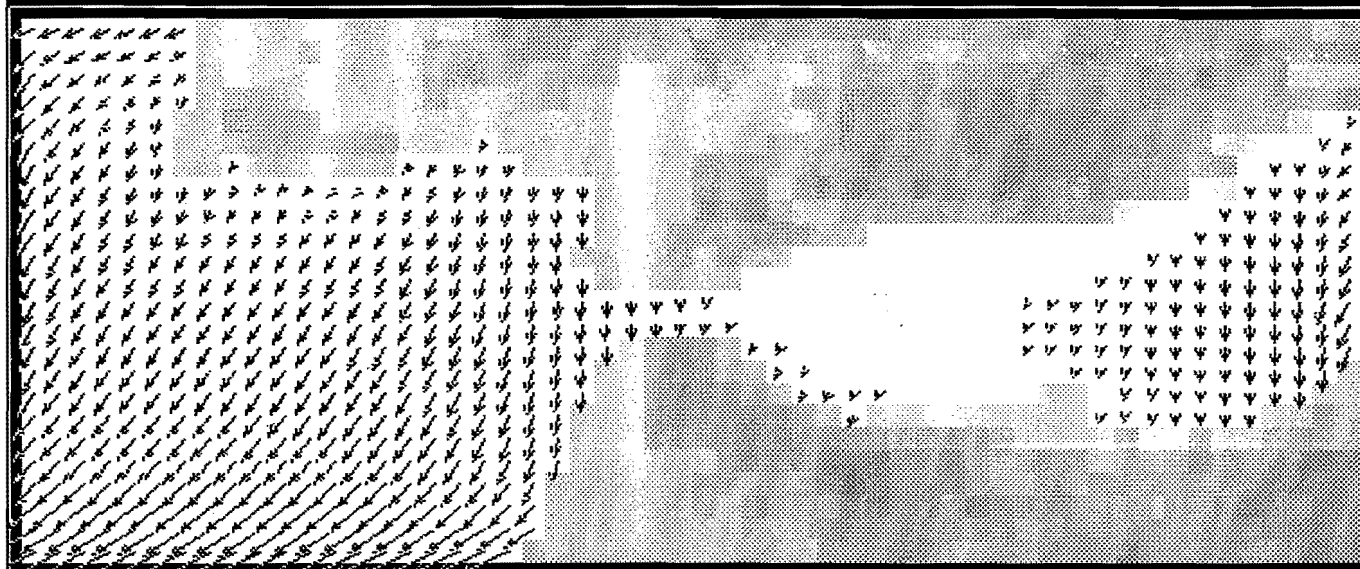
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = DA CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00 GAMMA = 0.0080

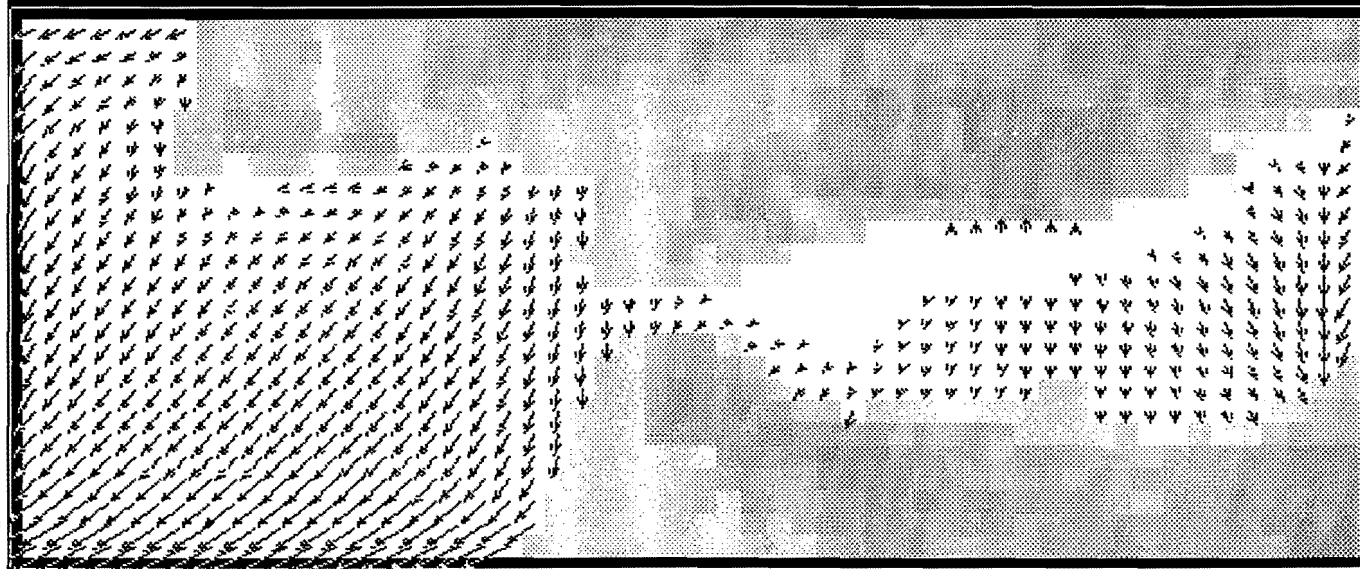
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.0014 CDLAND = 0.0200 ACSEA = 0.0400 ACLAND = 1.0000

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = EA CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00 GAMMA = 0.0040

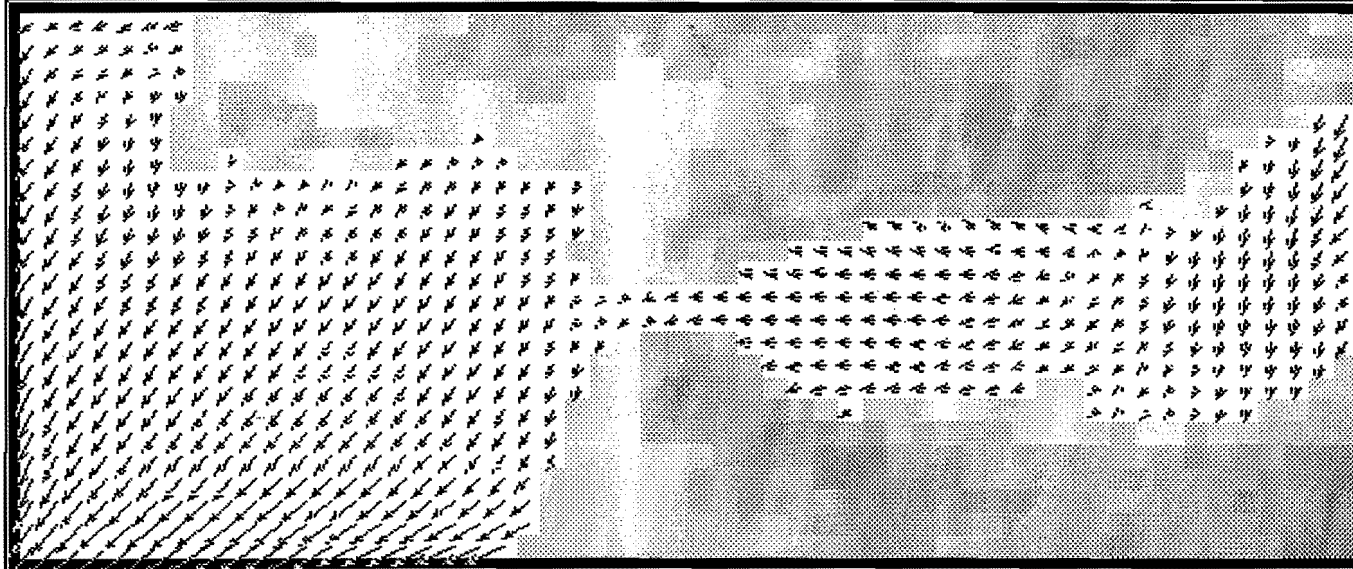
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.00 CDLAND = 0.02 ACSEA = 0.04 ACLAND = 1.00

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

FINISHED, PRESS A KEY

WAIT A MOMENT

DIA = 19 HORA = J ALCAN = EA CORIOL = T UGEOST = T TSUP = F

CROZ = 1.00 FACT = 1.00 EPSI = 1.00 GAMMA = 0.0080

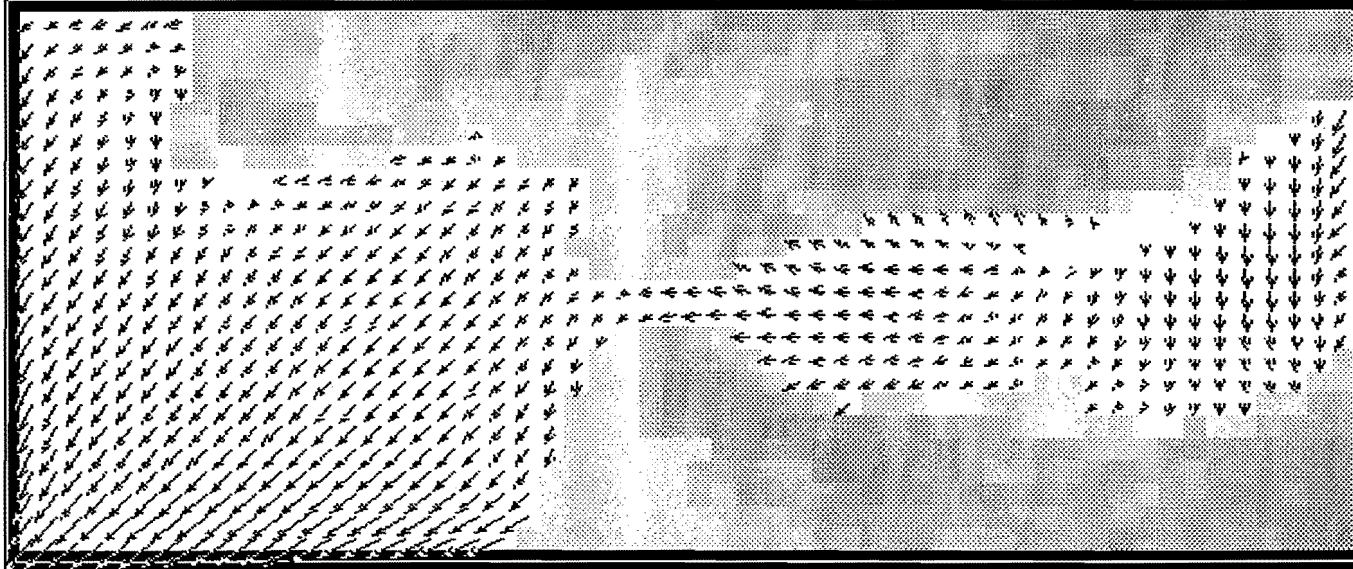
OROMAX = 4000.00 KM = 30000.00 KT = 300000.00

CDSEA = 0.0014 CDLAND = 0.0200 ACSEA = 0.0400 ACLAND = 1.0000

VIENTO ADAPTADO

20 M/S

50



PRESS A KEY TO CONTINUE

BLOQUE II

SITUACION PYREX

6^a Serie

**Efectos:
Rozamiento**

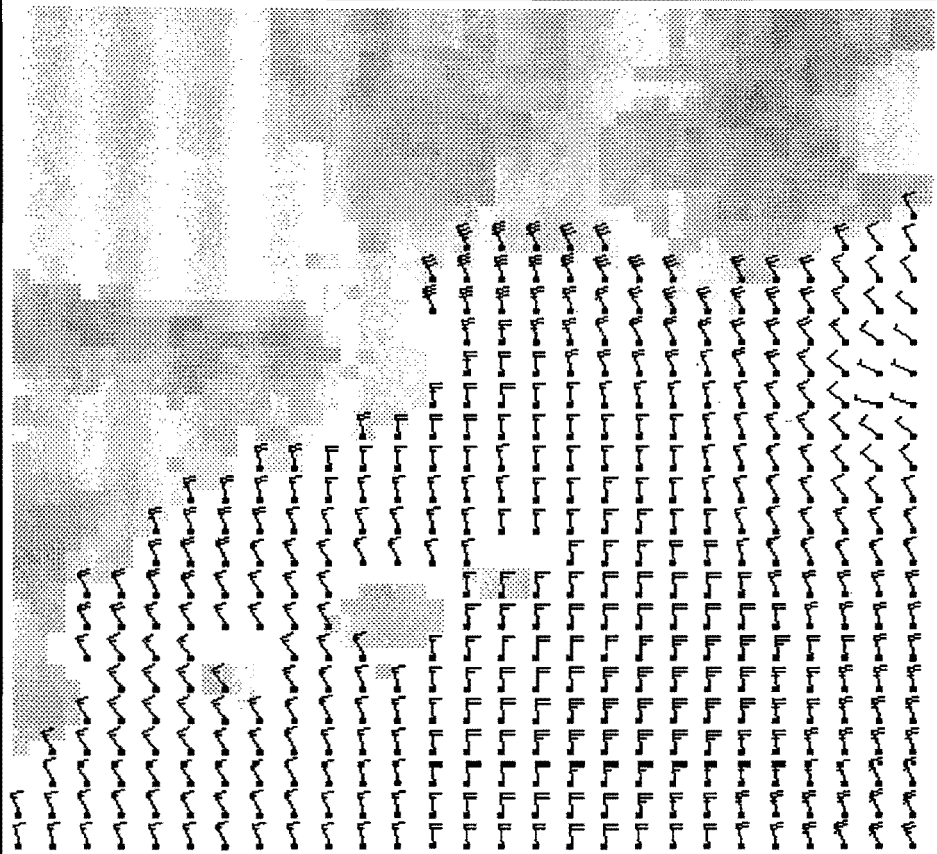
**Variables:
CROZ = 1**

VIENTO INICIAL



25 kt

0



DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 1.00

GAMMA = 0.0060

OROMAX = 4000.00

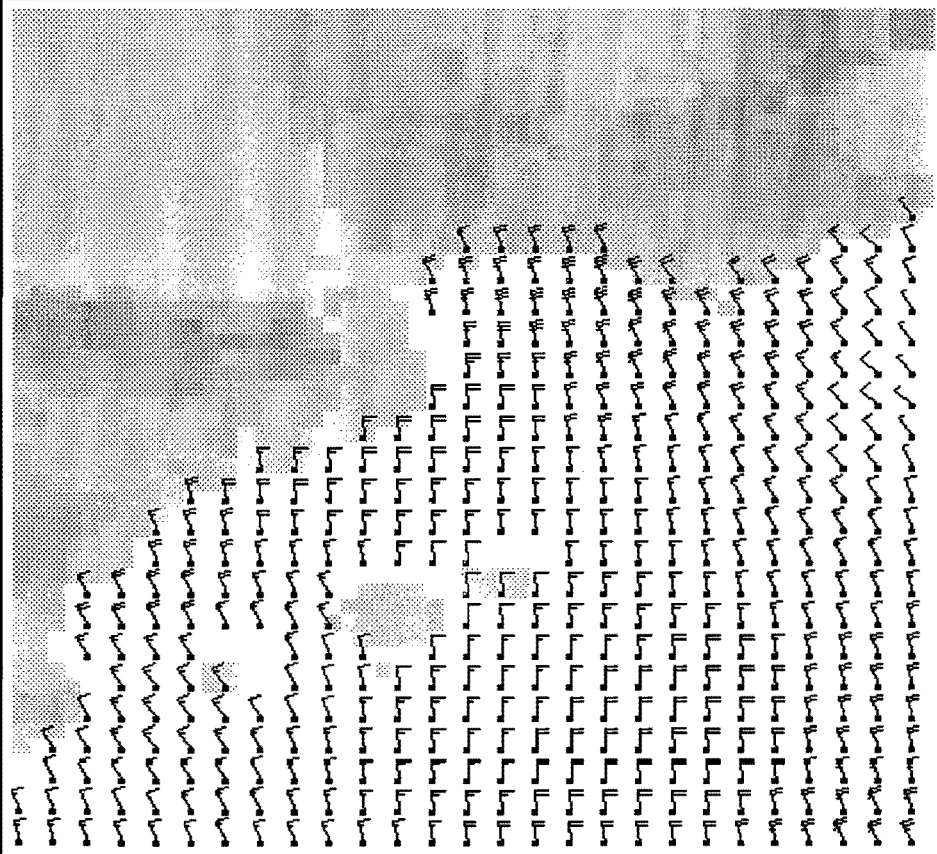
PRESS A KEY TO CONTINUE

VIENTO ADAPTADO



25 kt

10



DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 1.00

GAMMA = 0.0060

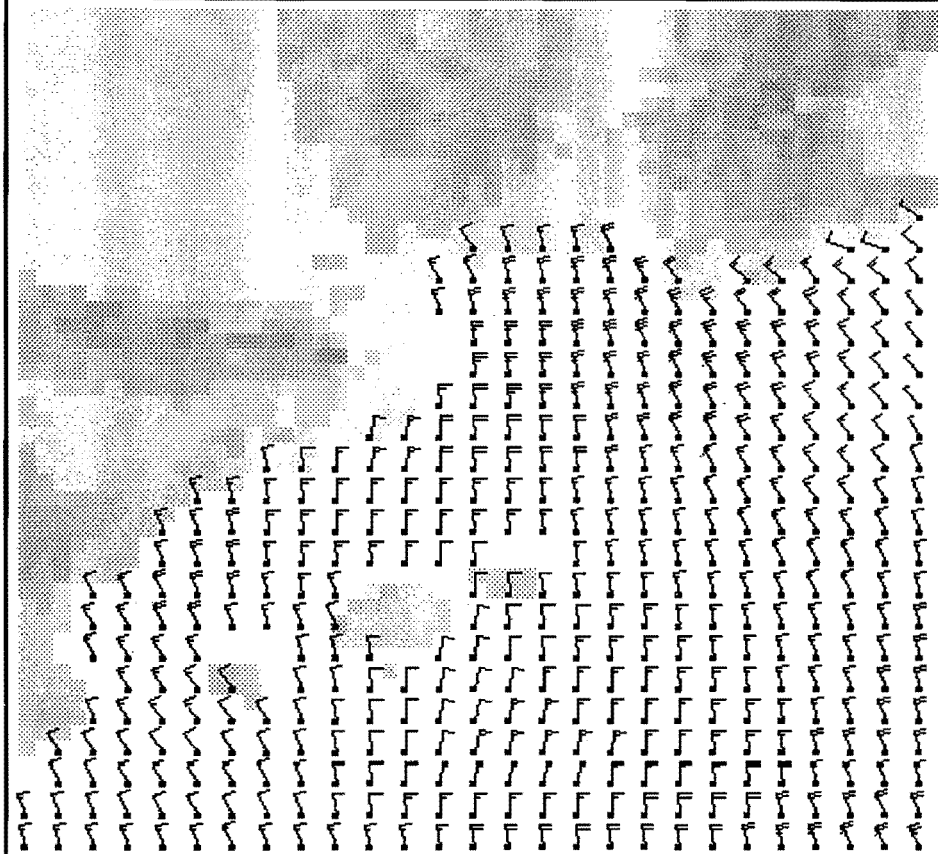
OROMAX = 4000.00

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

20



DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 1.00

GAMMA = 0.0060

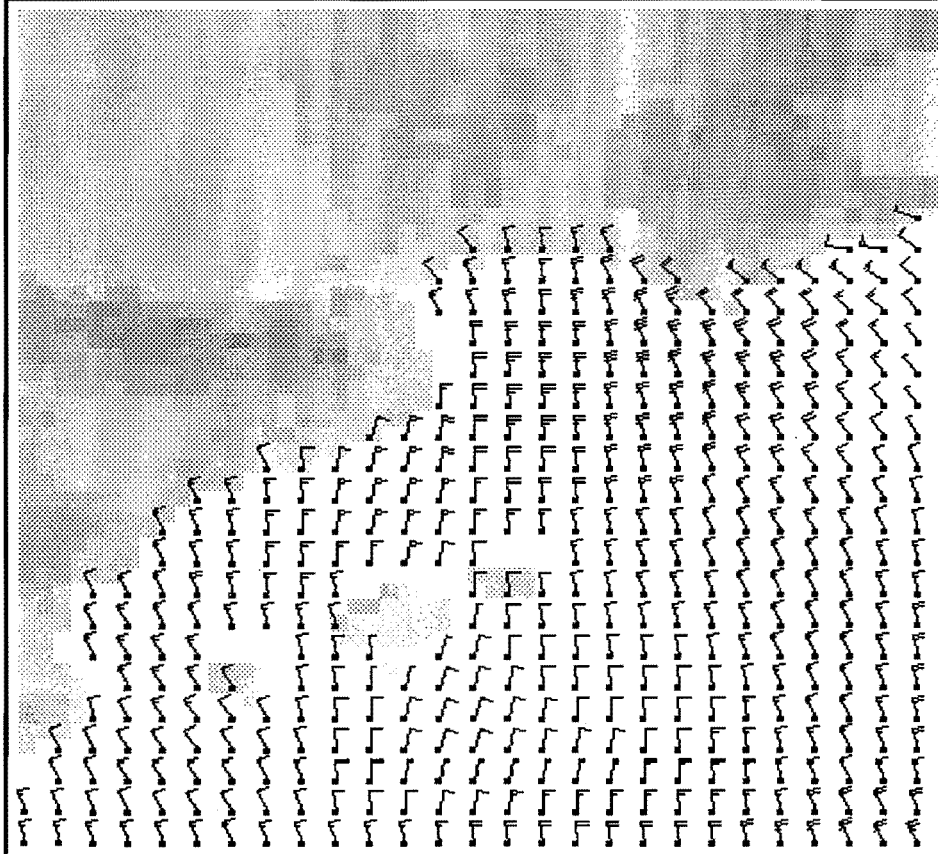
OROMAX = 4000.00

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

30



DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 1.00

GAMMA = 0.0060

OROMAX = 4000.00

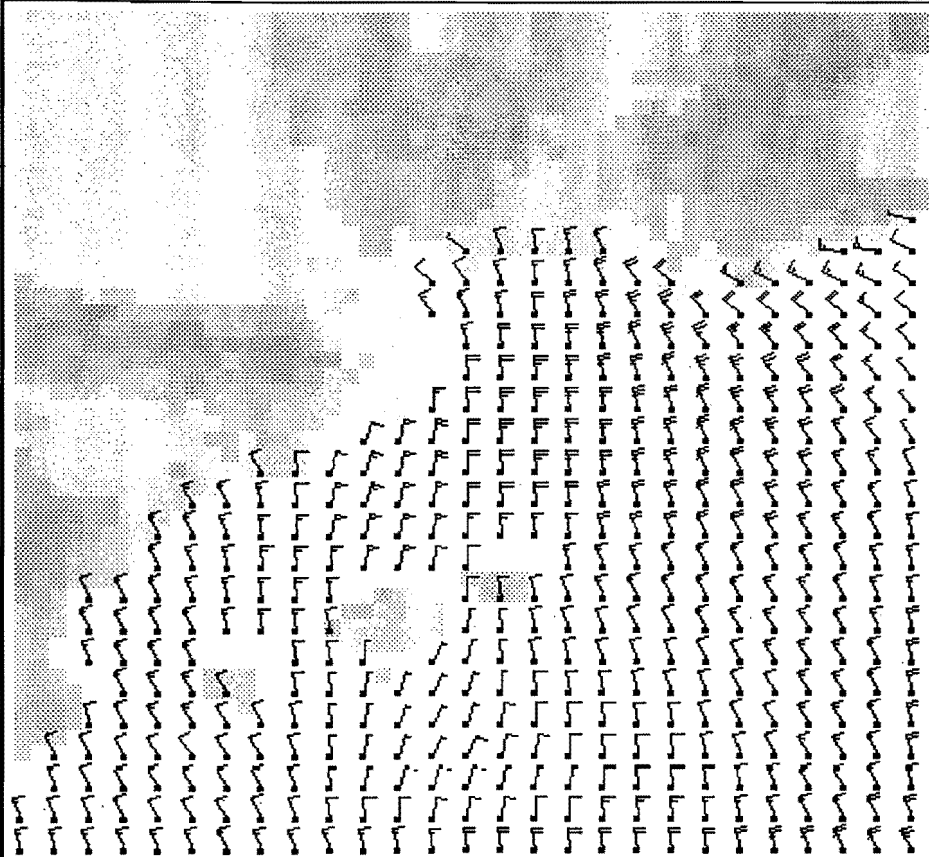
PRESS A KEY TO CONTINUE

VIENTO ADAPTADO



25 kt

40



DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 1.00

GAMMA = 0.0060

OROMAX = 4000.00

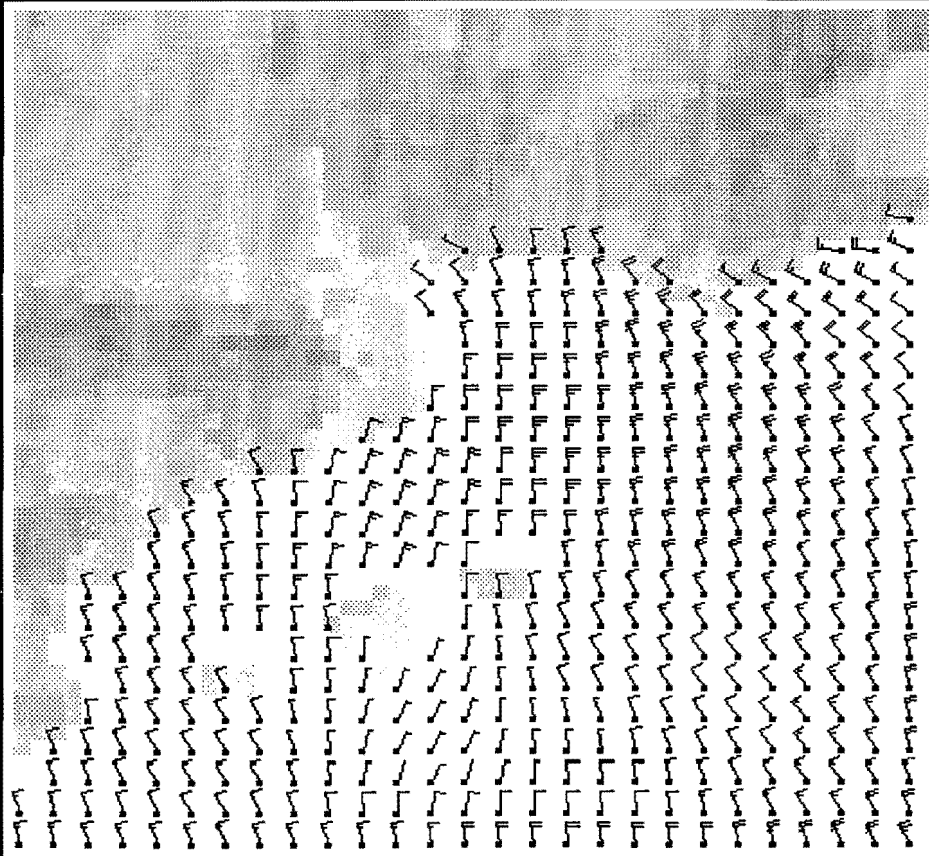
PRESS A KEY TO CONTINUE

VIENTO ADAPTADO



25 kt

50



DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 1.00

GAMMA = 0.0060

OROMAX = 4000.00

PRESS A KEY TO CONTINUE

7^a Serie

**Efectos:
Rozamiento**

**Variables:
CROZ = 1/2**

VIENTO INICIAL

25 kt

0

DIA = 15

HORA = K

ALCAN = AA

CORIOI = T

UGEOST = F

TSUP = F

CROZ = 0.50

GAMMA = 0.0060

OROMAX = 4000.00

PRESS A KEY TO CONTINUE

VIENTO ADAPTIADO

25 kt

10

DIA = 15

HORA = K

ALCAN = AA

CORIOI = T

UGEOST = F

TSUP = F

CROZ = 0.50

GAMMA = 0.0060

OROMAX = 4000.00

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

20

DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.50

GAMMA = 0.0060

OROMAX = 4000.00

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

30

DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.50

GAMMA = 0.0060

OROMAX = 4000.00

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

40

DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.50

GAMMA = 0.0060

ORDMAX = 4000.00

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

50

DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.50

GAMMA = 0.0060

ORDMAX = 4000.00

PRESS A KEY TO CONTINUE

8ª Serie

Efectos:
Rozamiento
Rozamiento en tierra

Variables:
CROZ = 1/5
CDLAND = 0.02

VIENTO INICIAL

25 kt

0

DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

PRESS A KEY TO CONTINUE

VIENTO ADAPTIADO

25 kt

10

DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

20

DIA = 15

HORA = K

ALCAN = AA

CORIOI = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

30

DIA = 15

HORA = K

ALCAN = AA

CORIOI = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

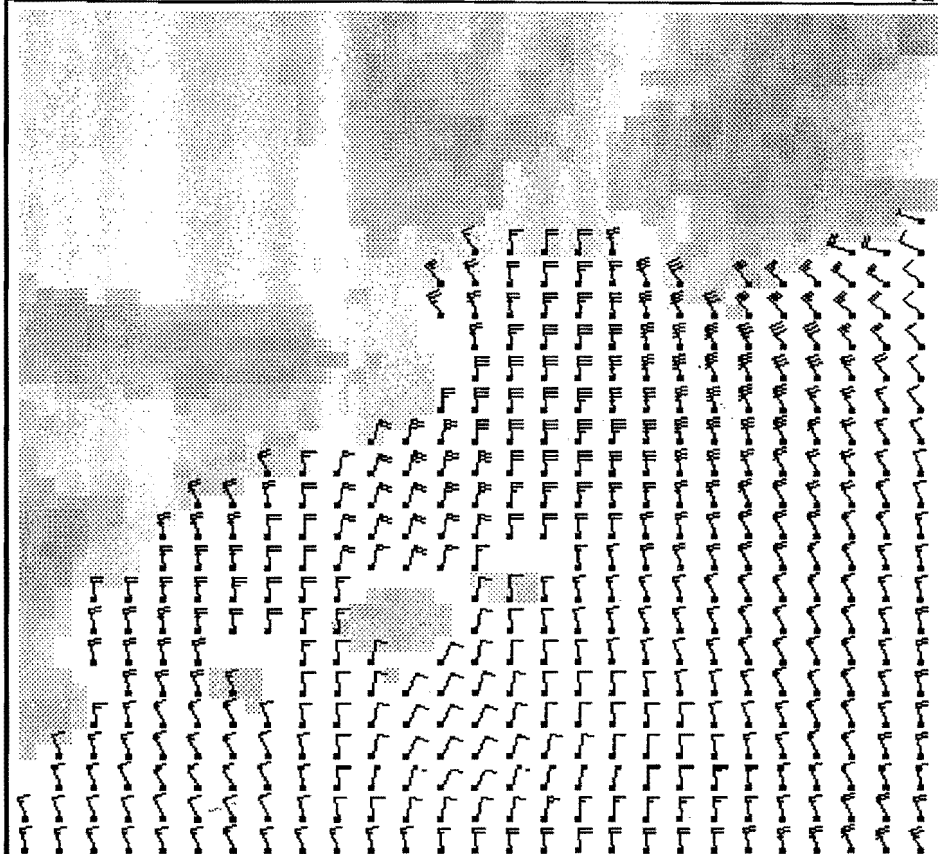
OROMAX = 4000.00

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

40



DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

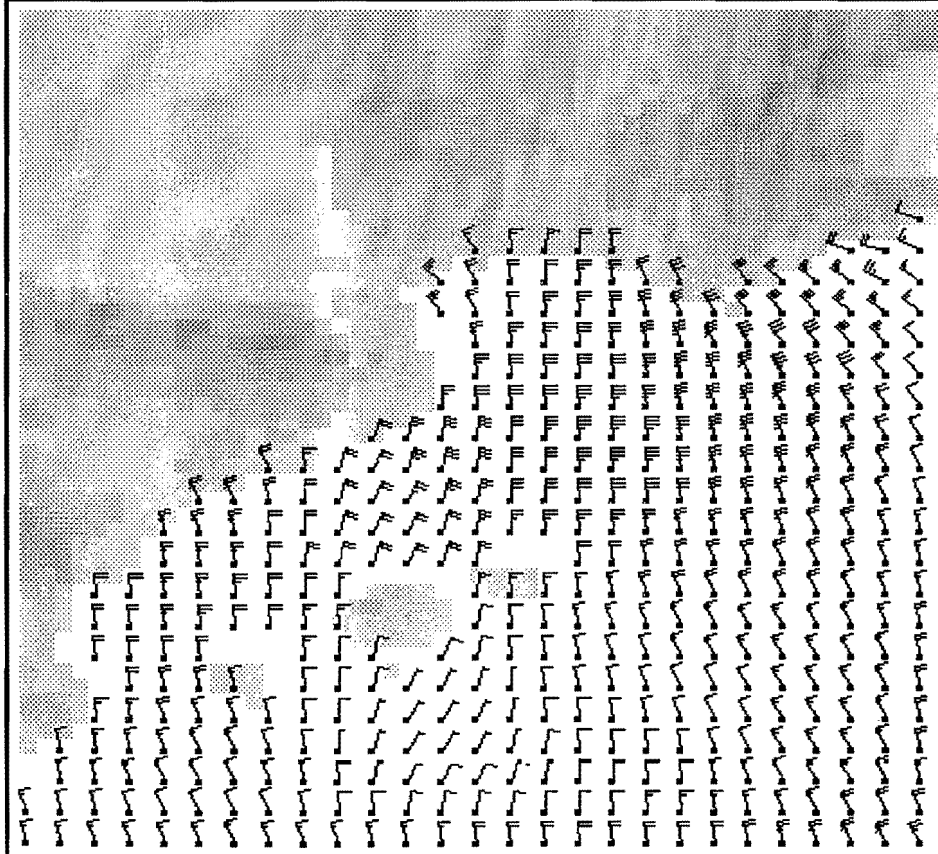
OROMAX = 4000.00

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

50



DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

PRESS A KEY TO CONTINUE

9ª Serie

Efectos:
Rozamiento
Rozamiento en tierra

Variables:
CROZ = 1/5
CDLAND = 0.01

VIENTO INICIAL

25 kt

0

DIA = 15

HORA = K

ALCAN = AA

CORIOI = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

CDLAND = 0.0100

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

10

DIA = 15

HORA = K

ALCAN = AA

CORIOI = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

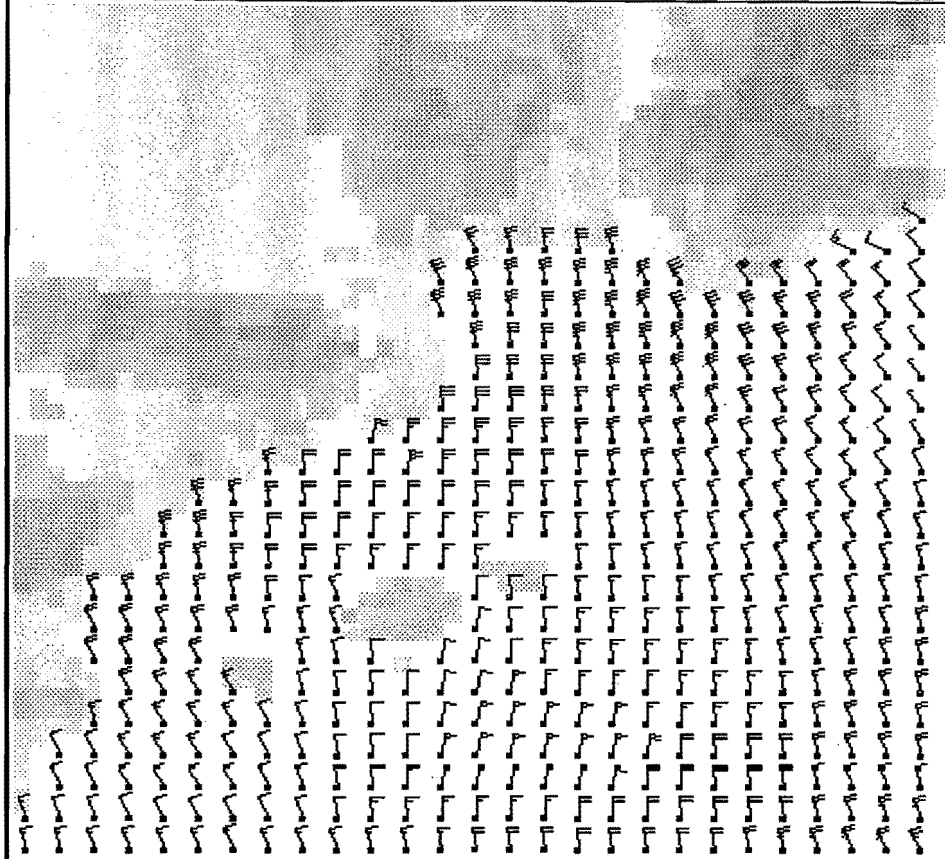
CDLAND = 0.0100

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

20



DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

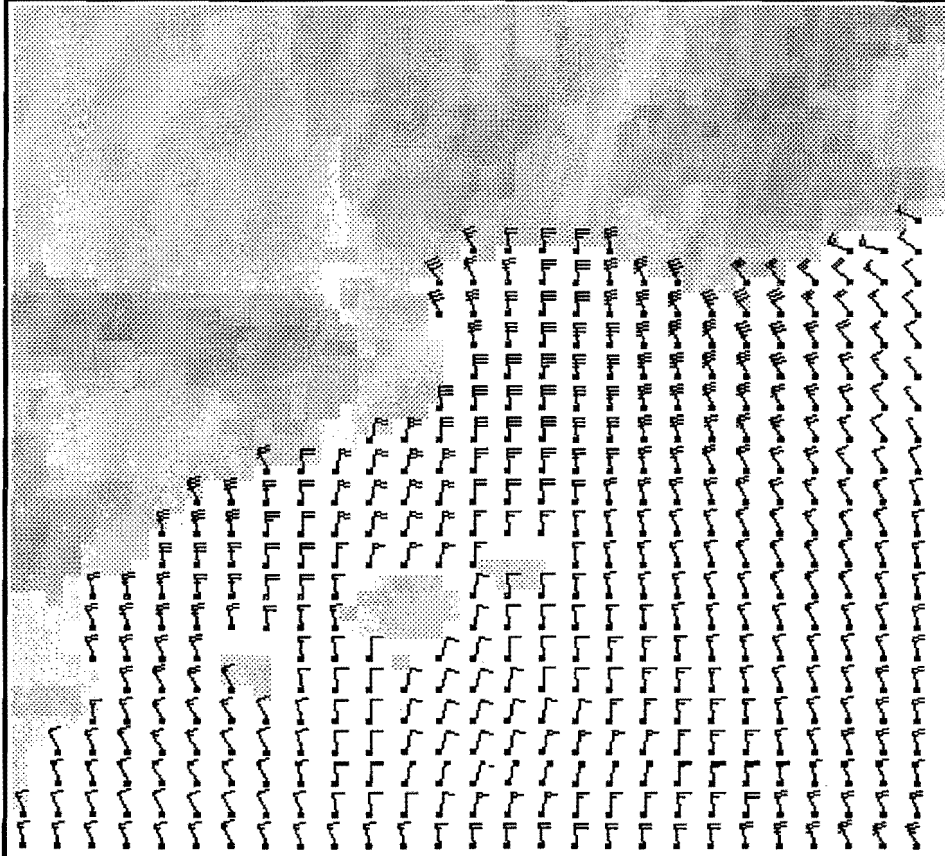
CDLAND = 0.0100

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

30



DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

CDLAND = 0.0100

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

40

DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

CDLAND = 0.0100

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

50

DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

CDLAND = 0.0100

PRESS A KEY TO CONTINUE

10^a Serie

**Efectos:
Rozamiento
Rozamiento en tierra**

**Variables:
CROZ = 1/5
CDLAND = 0.005**

VIENTO INICIAL

25 kt

0

DIA = 15

HORA = K

ALCAN = AA

CORIOI = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

CDLAND = 0.0050

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

10

DIA = 15

HORA = K

ALCAN = AA

CORIOI = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

CDLAND = 0.0050

PRESS A KEY TO CONTINUE

DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

CDLAND = 0.0050

PRESS A KEY TO CONTINUE

DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

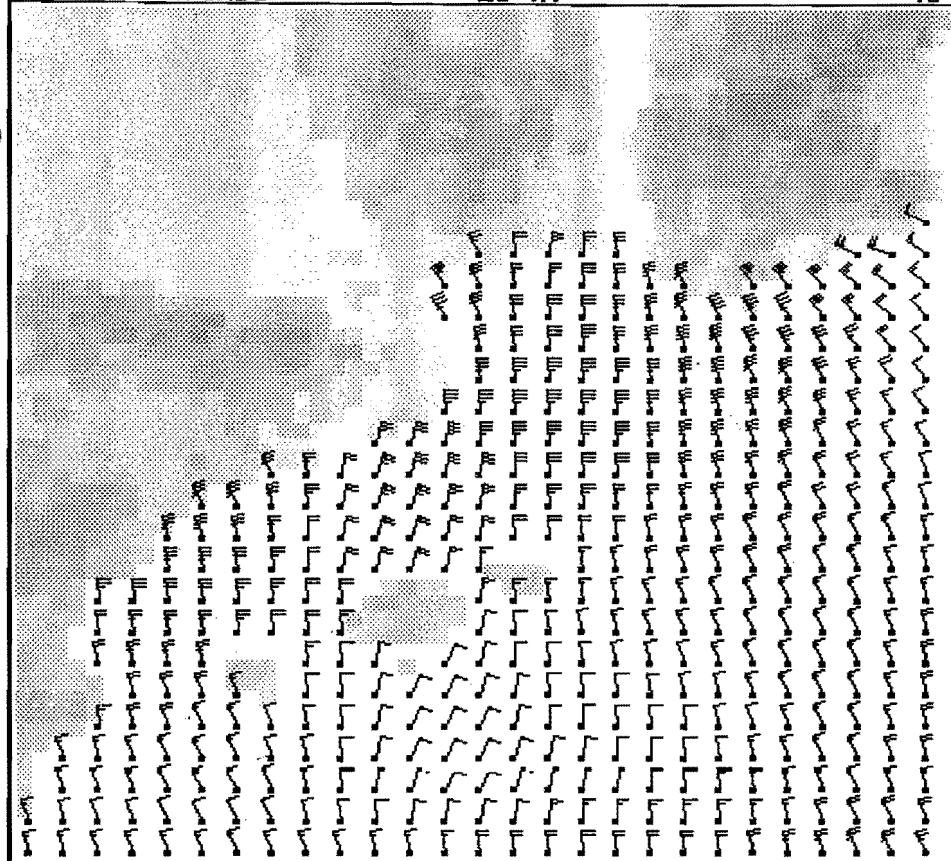
CDLAND = 0.0050

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

40



DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

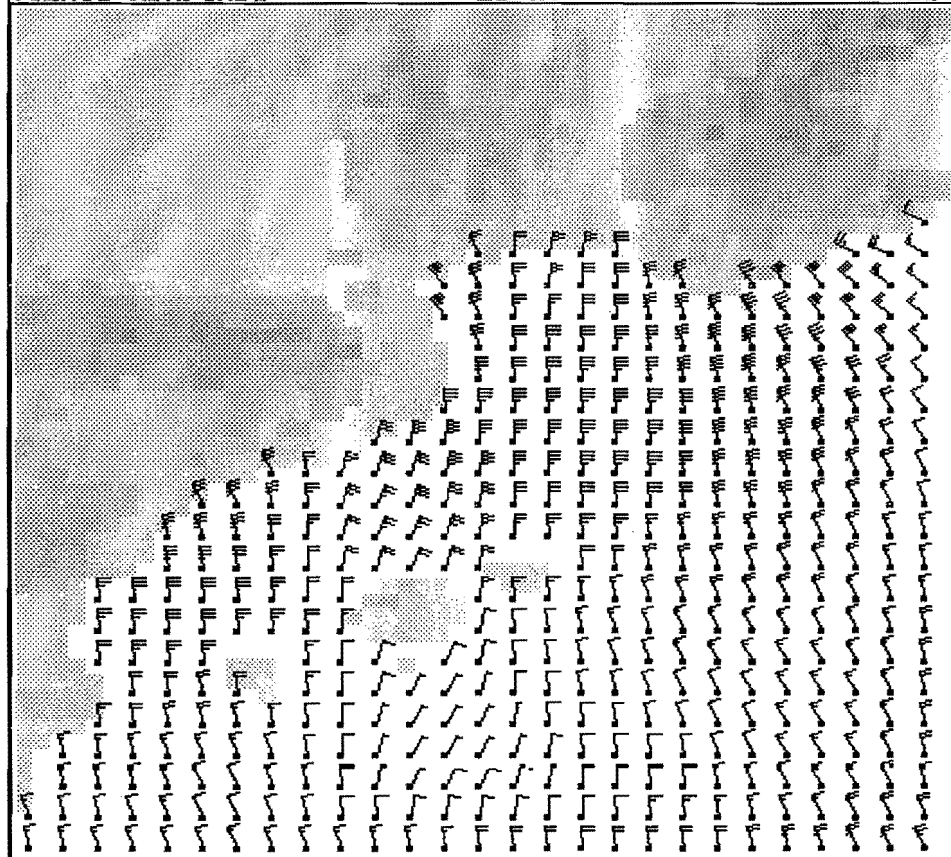
CDLAND = 0.0050

PRESS A KEY TO CONTINUE

VIENTO ADAPTADO

25 kt

50



DIA = 15

HORA = K

ALCAN = AA

CORIOL = T

UGEOST = F

TSUP = F

CROZ = 0.20

GAMMA = 0.0060

OROMAX = 4000.00

CDLAND = 0.0050

PRESS A KEY TO CONTINUE

APENDICE C
(Código del MASS
última versión)

```

PROGRAM MIS
PARAMETER(LG=54,MG=54)
INCLUDE 'GREX.FH'
COMMON /A01/ L, M, L1, M1, M2, G,
1 SLAT, DPHI
COMMON /A02/ NSTEP, DT, DUSMIN, ALFA(LG,MG),ISTEP, NADIAB
COMMON /A03/ DX(MG), COR(MG), DY
COMMON /A04/ ICAB(30),TNDTEM,TNDCOU,TNDCOV
COMMON /A05/ ZS(LG,MG), ZR(LG,MG), U(LG,MG), V(LG,MG),
1 TR(LG,MG), TS(LG,MG), TH(LG,MG), CD(LG,MG),
1 AC(LG,MG)
COMMON /A06/ A(LG,MG), B(LG,MG), C(LG,MG)
COMMON /A07/ A1(LG,MG), A2(LG,MG), A3(LG,MG), C1(LG,MG),
1 C2(LG,MG), E1(LG,MG), E2(LG,MG)
COMMON /A08/ GAMMAD, GAMMA, CP,
1 KM, KT, H, RNU, SMOROG, FACTOR, OROMAX,TSUP,
1 CDSEA,CDLAND,CROZ,ACSEA,ACLAND,VGEOST,EPSI
COMMON /A09/ TENDT(LG,MG),TENDU(LG,MG),TENDV(LG,MG)
COMMON /A10/ TENDT1(LG,MG),TENDU1(LG,MG),TENDV1(LG,MG)
COMMON /A11/ BOUNDT(LG,MG),BOUNDU(LG,MG),BOUNDV(LG,MG)
COMMON /A12/ DIA, HORA,ALCAN,REFLEV
COMMON /GRAF/ LIMX,LIMY,MAX_COLOR,MODE,ICON,CORIOL
NAMELIST/DATOS/DIA,HORA,ALCAN,REFLEV,
+SLAT,DPHI,KT,KM,RNU,SMOROG,FACTOR,OROMAX,
+NSTEP,NADIAB,DT,DUSMIN,GAMMA,TSUP,VGEOST,CORIOL,
+CDSEA,CDLAND,CROZ,EPSI,
+ACSEA,ACLAND,SIZE,ANCHO,ISKIPP
INTEGER LIMX,LIMY,MAX_COLOR
INTEGER MODE
INTEGER VIDEO_SYSTEM(4)
CHARACTER*10 STRING
CHARACTER*2 DIA,ALCAN,REFLEV
CHARACTER*1 HORA
CHARACTER*8 ELEM
LOGICAL TSUP,VGEOST,CORIOL
REAL KT,KM
DATA SLAT/40./
DATA DPHI/0.166/
DATA KT/30000./
DATA KM/30000./
DATA RNU/0.5/
DATA SMOROG/0.2/
DATA FACTOR/1./
DATA NSTEP/10/
DATA NADIAB/10/
DATA DT/240./
DATA DUSMIN/0.000001/
DATA GAMMA/0.006/
DATA TSUP/.FALSE./
DATA VGEOST/.TRUE./
DATA CDSEA/0.0014/
DATA CDLAND/0.02/
DATA CROZ/1./
DATA EPSI/0./
DATA ACSEA/0.04/
DATA ACLAND/1./
DATA SIZE/10./
DATA ANCHO/10./

```

```

DATA ISKIPP/1/
DATA ICON/0/
MODE=VIDEO_CONFIGURATION(VIDEO_SYSTEM)
IER=GRAPHICS_MODE(MODE)
IER=GET_DEVICE_LIMITS(LIMX,LIMY,MAX_COLOR)
IER=GRAPHIC_TEXT('MASS & DEMPSEY MODEL      ', 10,LIMY-20,14)
IER=GRAPHIC_TEXT('PRESS A KEY TO CONTINUE',10,LIMY-10,14)
IER=PAUSE()
IER=CLEAR()
N=0
IER=SET_COLOR(N)
IER=TEXT_MODE()
M1 = MG-1
L1 = LG-1
M2 = MG-2
L2 = LG-2
M3 = MG-3
L3 = LG-3
M4 = MG-4
L4 = LG-4
M5 = MG-5
L5 = LG-5
M6 = MG-6
L6 = LG-6
M7 = MG-7
L7 = LG-7
M8 = MG-8
L8 = LG-8
G = 9.81
CP=1004.6
GAMMAD=G/CP
H=2000.
PI = 4.*ATAN(1.)
RAD = PI/180.
DY = 111.1*DPHI*1.E03

```

C

C LEE OPCIONES DEL NAMELIST

C

```

PRINT*, ' READ INFORMATION IN \PRNUJAR\FORTRAN\MIS.INT'
OPEN(8,FILE='C:\PRNUJAR\FORTRAN\MIS.INT')
READ(8,DATOS)
WRITE(6,DATOS)
DO 50 J=1,MG
  X = (SLAT+(J-1)*DPHI)*RAD
  DX(J) = DY*COS(X)
  COR(J) = 2.*7.272E-05*SIN(X)
50 CONTINUE

```

```

DO 60 J=1,MG
  DO 60 I=1,LG
    IF(I.EQ.1.OR.I.EQ.LG.OR.J.EQ.1.OR.J.EQ.MG)THEN
      ALFA(I,J)=1.0
    ELSEIF(I.EQ.2.OR.I.EQ.L1.OR.J.EQ.2.OR.J.EQ.M1)THEN
C      ALFA(I,J)=0.538
      ALFA(I,J)=0.0
    ELSEIF(I.EQ.3.OR.I.EQ.L2.OR.J.EQ.3.OR.J.EQ.M2)THEN
C      ALFA(I,J)=0.238
      ALFA(I,J)=0.0
    ELSEIF(I.EQ.4.OR.I.EQ.L3.OR.J.EQ.4.OR.J.EQ.M3)THEN
C      ALFA(I,J)=0.095
      ALFA(I,J)=0.0
    ELSE
      ALFA(I,J)=0.0
    ENDIF
  60 CONTINUE
C
C LEE LA OROGRAFIA.
C
  ELEM='OROGRAPH'
  CALL IBUF(30,ZS,LG,MG,ELEM,IERR)
  DO 130 I=1,LG
    DO 130 J=1,MG
      ZS(I,J)=ZS(I,J)*FACTOR
      IF(ZS(I,J).GT.ROMAX) ZS(I,J)=ROMAX
      IF(ZS(I,J).GT. 2.0*FACTOR) THEN
        AC(I,J)=ACLAND/3600.
        CD(I,J)=CDLAND
      ELSE
        CD(I,J)=CDSEA
        AC(I,J)=ACSEA/3600.
      ENDIF
    130 CONTINUE
  C  CALL SMOOTH(ZS,B,LG,MG,SMOROG)
  C
  C LEE EL GEOPOTENCIAL (ZR) Y LA TEMPERATURA (TR) DE REFERENCIA.
  C
    ELEM(2:2)=HORA
    ELEM(3:4)=ALCAN
    ELEM(5:6)=REFLEV
    ELEM(7:8)=DIA
    ELEM(1:1)='A'
    CALL IBUF(50,ZR,LG,MG,ELEM,IERR)
    ELEM(1:1)='B'
    CALL IBUF(51,TR,LG,MG,ELEM,IERR)
    DO 140 I=1,LG
      DO 140 J=1,MG
        ZR(I,J)=ZR(I,J)*10.
    140 CONTINUE

```

```

C
C CALCULA LA TEMPERATURA EN SUPERFICIE
C
  DO 100 I=1, LG
    DO 100 J=1, MG
      TS(I,J) = TR(I,J) + GAMMA * (ZR(I,J) - ZS(I,J))
      TH(I,J) = TR(I,J) + GAMMA * (ZR(I,J) - ZS(I,J) - H)
100  CONTINUE
    IF(TSUP) THEN
      ELEM(1:1) = 'B'
      ELEM(5:6) = '99'
      CALL IBUF(52, TS, LG, MG, ELEM, IERR)
    ENDIF
    IF(VGEOST) THEN
C
C CALCULA EL VIENTO INICIAL A PARTIR DEL EQUILIBRIO ENTRE LA FUERZA DE
C PRESION, DE CORIOLIS Y DE ROZAMIENTO.
C
      CALL WINIT
      GOTO 101
    ENDIF

C
C LEE LAS COMPONENTES ZONAL Y MERIDIANA DEL VIENTO EN 1000MB.
C
  ELEM(5:6) = '85'
  ELEM(1:1) = 'I'
  CALL IBUF(40, B, LG, MG, ELEM, IERR)
  ELEM(1:1) = 'J'
  CALL IBUF(41, A, LG, MG, ELEM, IERR)
101 CONTINUE
  IER = GRAPHICS_MODE(MODE)
  IER = SET_COLOR(15)
  IER = FILLED_RECTANGLE(0, 0, 700, 700)
  CALL DIBWIN(B, A, ZS, LG, MG, SIZE, ANCHO, ISTEP, ISKIPP, IERR)
  IER = GRAPHIC_TEXT('WAIT A MOMENT', 450, LIMY-450, 0)

C
C DEFINE U Y V EN LA REJILLA ALTERNADA C DE ARAKAWA
C
  DO 110 J=1, MG
    DO 110 I=1, LG
      IF(I.NE.LG) THEN
        U(I,J) = 0.5 * (B(I,J) + B(I+1,J))
      ELSE
        U(I,J) = 1.5 * B(I,J) - B(I-1,J)
      ENDIF
      IF(J.NE.MG) THEN
        V(I,J) = 0.5 * (A(I,J) + A(I,J+1))
      ELSE
        V(I,J) = 1.5 * A(I,J) - A(I,J-1)
      ENDIF
    110 CONTINUE

```

```

DO 150 I=1, LG
  DO 150 J=1, MG
    BOUNDT(I,J)=TS(I,J)
    BOUNDU(I,J)=U(I,J)
    BOUNDV(I,J)=V(I,J)
150 CONTINUE

  ISTEP=0
  DO 10 K=1, NSTEP
    CALL FORCST
    CALL TSTEP
    CALL BOUND
    IF(ABS(TNDCOU).LT.DUSMIN.OR.ABS(TNDCOV).LT.DUSMIN) GOTO 11
    ISTEP=ISTEP+1
    IF(MOD(ISTEP,10).EQ.0) THEN
      IER= CLEAR()
C
C  DEFINE U Y V EN LA REJILLA ORIGINARIA.
C
      DO 12 J=1, MG
        DO 12 I=1, LG
          IF(I.NE.1) THEN
            A(I+1,J)=.5*(U(I,J)+U(I+1,J))
          ELSE
            A(I,J)=U(1,J)+0.5*(U(1,J)-U(2,J))
          ENDIF
          IF(J.NE.1) THEN
            B(I,J+1)=0.5*(V(I,J)+V(I,J+1))
          ELSE
            B(I,J)=V(I,1)+0.5*(V(I,1)-V(I,2))
          ENDIF
12 CONTINUE
        IER=SET_COLOR(15)
        IER=FILLED_RECTANGLE(0,0,700,700)
        CALL DIBWIN(U,V,ZS, LG, MG, SIZE, ANCHO, ISTEP, ISKIPP, IERR)
        IER=GRAPHIC_TEXT('WAIT A MOMENT',450,LIMY-450,0)
      ENDIF
10 CONTINUE
11 CONTINUE
    IER=GRAPHIC_TEXT('FINISHED, PRESS A KEY ',450,LIMY-470,0)
    IER=PAUSE()
    IER= CLEAR()
    IER=TEXT_MODE()
    STOP 'O.K.'
  END

```


SUBROUTINE WINIT

C

```

*****
*                                     *
* FUNCTION * THIS ROUTINE COMPUTES THE INITIAL WIND. *
*                                     *
*****

PARAMETER(LG=54,MG=54)
COMMON /A01/ L, M, L1, M1, M2, G,
1 SLAT, DPHI
COMMON /A02/ NSTEP, DT, DUSMIN, ALFA(LG,MG),ISTEP,NADIAB
COMMON /A03/ DX(MG), COR(MG), DY
COMMON /A04/ ICAB(30),TNDTEM,TNDCOU,TNDCOV
COMMON /A05/ ZS(LG,MG), ZR(LG,MG), U(LG,MG), V(LG,MG),
1 TR(LG,MG), TS(LG,MG), TH(LG,MG), CD(LG,MG),
1 AC(LG,MG)
COMMON /A06/ A(LG,MG), B(LG,MG), C(LG,MG)
COMMON /A07/ A1(LG,MG), A2(LG,MG), A3(LG,MG), C1(LG,MG),
1 C2(LG,MG), E1(LG,MG), E2(LG,MG)
COMMON /A08/ GAMMAD, GAMMA, CP,
1 KM, KT, H, RNU, SMOROG, FACTOR, OROMAX,TSUP,
1 CDSEA,CDLAND,CROZ,ACSEA,ACLAND,VGEOST,EPSI
COMMON /A09/ TENDT(LG,MG),TENDU(LG,MG),TENDV(LG,MG)
COMMON /A10/ TENDT1(LG,MG),TENDU1(LG,MG),TENDV1(LG,MG)
COMMON /A11/ BOUNDT(LG,MG),BOUNDU(LG,MG),BOUNDV(LG,MG)
COMMON /GRAF/ LIMX,LIMY,MAX_COLOR,MODE,CORIOL
REAL KM,KT
LOGICAL TSUP,VGEOST
ITER=1
IF(I1.EQ.1)ITER=0
DO 150 J=2,MG-1
DO 150 I=2,LG-1
A(I,J)=+(1/COR(J))*
+ (+G*(ZR(I+1,J)-ZR(I-1,J))/(DX(J)**2.)
+ -G*(ZR(I,J)-ZS(I,J))*(TR(I+1,J)-TR(I-1,J))/(TR(I,J)**2.*DX(J))
+ +ITER*EPSI*(CD(I,J)*CROZ*2.8*U(I,J)*SQRT((ABS(U(I,J)))**2.+
+ ABS(V(I,J))**2.)/H))
B(I,J)=-{(1/COR(J))*
+ (+G*(ZR(I,J+1)-ZR(I,J-1))/(DY**2.)
+ -G*(ZR(I,J)-ZS(I,J))*(TR(I,J+1)-TR(I,J-1))/(TR(I,J)*DY**2.)
+ +ITER*EPSI*(CD(I,J)*CROZ*2.8*V(I,J)*SQRT(ABS(U(I,J))**2.+
+ ABS(V(I,J))**2.)/H))
150 CONTINUE
DO 152 I=1,LG
A(I,1)=2.*A(I,2)-A(I,3)
B(I,1)=2.*B(I,2)-B(I,3)
A(I,MG)=2.*A(I,MG-1)-A(I,MG-2)
B(I,MG)=2.*B(I,MG-1)-B(I,MG-2)
152 CONTINUE
DO 153 J=1,MG
A(1,J)=2.*A(2,J)-A(3,J)
B(1,J)=2.*B(2,J)-B(3,J)
A(LG,J)=2.*A(LG-1,J)-A(LG-2,J)
B(LG,J)=2.*B(LG-1,J)-B(LG-2,J)
153 CONTINUE

```

SUBROUTINE FORCST

```

C
*****
*
* FUNCTION * THIS ROUTINE COMPUTES THE TENDENCIES.
*
*****

PARAMETER(LG=54,MG=54)
COMMON /A01/ L, M, L1, M1, M2, G,
1 SLAT, DPHI
COMMON /A02/ NSTEP, DT, DUSMIN, ALFA(LG,MG),ISTEP,NADIAB
COMMON /A03/ DX(MG), COR(MG), DY
COMMON /A04/ ICAB(30),TNDTEM,TNDCOU,TNDCOV
COMMON /A05/ ZS(LG,MG), ZR(LG,MG), U(LG,MG), V(LG,MG),
1 TR(LG,MG), TS(LG,MG), TH(LG,MG), CD(LG,MG),
1 AC(LG,MG)
COMMON /A06/ A(LG,MG), B(LG,MG), C(LG,MG)
COMMON /A07/ A1(LG,MG), A2(LG,MG), A3(LG,MG), C1(LG,MG),
1 C2(LG,MG), E1(LG,MG), E2(LG,MG)
COMMON /A08/ GAMMAD, GAMMA, CP,
1 KM, KT, H, RNU, SMOROG, FACTOR, OROMAX,TSUP,
1 CDSEA,CDLAND,CROZ,ACSEA,ACLAND,VGEOST,EPSI
COMMON /A09/ TENDT(LG,MG),TENDU(LG,MG),TENDV(LG,MG)
COMMON /A10/ TENDT1(LG,MG),TENDU1(LG,MG),TENDV1(LG,MG)
COMMON /A11/ BOUNDT(LG,MG),BOUNDU(LG,MG),BOUNDV(LG,MG)
COMMON /GRAF/ LIMX,LIMY,MAX_COLOR,MODE,CORIOL
REAL KM,KT
LOGICAL TSUP,VGEOST,CORIOL
DO 200 I=1,LG
DO 200 J=1,MG
GAMMA2=(TS(I,J)-TH(I,J))/H
C1(I,J)=TS(I,J)/TH(I,J)
C2(I,J)=TS(I,J)*ALOG(C1(I,J))/(GAMMA2*H)
A1(I,J)=1.-(GAMMAD/GAMMA2)*(1.-C2(I,J))
A2(I,J)=GAMMAD*(C1(I,J)*(1./GAMMA2-1./GAMMA)-C2(I,J)/
+ GAMMA2)
A3(I,J)=(GAMMAD/GAMMA)*(TS(I,J)/TR(I,J))
E1(I,J)=(TS(I,J)/GAMMA2)*(1./TH(I,J)-ALOG(TS(I,J)/TH(I,J))/
+ (H*GAMMA2))
E2(I,J)=TS(I,J)/TH(I,J)
200 CONTINUE
CALL LAPSIG(B,TS,DX,DY,LG,MG)
CALL LAPSIG(A,U,DX,DY,LG,MG)
CALL LAPSIG(C,V,DX,DY,LG,MG)

C
C CALCULO TENDENCIA DE LA TEMPERATURA DEL SUELO.
C
DO 210 J=2,MG-1
DO 210 I=2,LG-1
C ADVECCION TEMPERATURA SUELO
TENDT(I,J)=
+ -0.5*(U(I,J)+U(I-1,J))*(TS(I+1,J)-TS(I-1,J))/(2.*DX(J))
TENDT(I,J)=TENDT(I,J)
+ -0.5*(V(I,J)+V(I,J-1))*(TS(I,J+1)-TS(I,J-1))/(2.*DY)
C ADVECCION TEMPERATURA NIVEL H
TENDT(I,J)=TENDT(I,J)
+ -0.5*(U(I,J)+U(I-1,J))*(A2(I,J)/A1(I,J))
+ *(TH(I+1,J)-TH(I-1,J))/(2.*DX(J))

```

```

DIFU=0.
DIFV=0.
DO 151 J=1, MG
  DO 151 I=1, LG
    DIFU=DIFU+ABS(B(I,J)-U(I,J))
    DIFV=DIFV+ABS(A(I,J)-V(I,J))
151 CONTINUE
DIFU=DIFU/(MG*LG)
DIFV=DIFV/(MG*LG)
DO 154 J=1, MG
  DO 154 I=1, LG
    U(I,J)=B(I,J)
    V(I,J)=A(I,J)
154 CONTINUE
IF(EPSI.EQ.0.)THEN
  PRINT*, ' EPSI=0--> CALCULA VIENTO INICIAL GEOSTROFICO'
  RETURN
ENDIF
149 CONTINUE
PRINT*, 'END COMPUTATION OF INITIAL GRADIENT WIND'
RETURN
END

```

```

        TENDT(I,J) = TENDT(I,J)
+       -0.5*(V(I,J) + V(I,J-1)) * (A2(I,J)/A1(I,J))
+       *(TH(I,J+1)-TH(I,J-1))/(2.*DY)
C  ADVECCION TEMPERATURA NIVEL REFERENCIA
        TENDT(I,J) = TENDT(I,J)
+       -0.5*(U(I,J) + U(I-1,J)) * (A3(I,J)/A1(I,J))
+       *(TR(I+1,J)-TR(I-1,J))/(2.*DX(J))
        TENDT(I,J) = TENDT(I,J)
+       -0.5*(V(I,J) + V(I,J-1)) * (A3(I,J)/A1(I,J))
+       *(TR(I,J+1)-TR(I,J-1))/(2.*DY)
C  DIFUSION EN LA HORIZONTAL DE LA TEMPERATURA
        TENDT(I,J) = TENDT(I,J) + KT*B(I,J)/A1(I,J)
C  DIABATIC FORCING
        IF(ISTEP.GT.NADIAB)
+       TENDT(I,J) = TENDT(I,J) + AC(I,J)/A1(I,J)
C
C  CALCULO TENDENCIA DE LA COMPONENTE ZONAL DEL VIENTO EN EL SUELO
C
C  ADVECCION COMPONENTE ZONAL
        TENDU(I,J) =
+       -U(I,J)*(U(I+1,J)-U(I-1,J))/(2.*DX(J))
        TENDU(I,J) = TENDU(I,J)
+       -0.25*(V(I,J) + V(I,J-1) + V(I+1,J) + V(I+1,J-1)) *
+       (U(I,J+1)-U(I,J-1))/(2.*DY)
C  TERMINO DE CORIOLIS
        IF(CORIOL) TENDU(I,J) = TENDU(I,J)
+       + COR(J)*0.25*(V(I,J) + V(I,J-1) + V(I+1,J) + V(I+1,J-1))
C  GRADIENTE TEMPERATURA SUPERFICIE
        TENDU(I,J) = TENDU(I,J)
+       -G*0.5*((E1(I,J)-H/TH(I,J)) + (E1(I+1,J)-H/TH(I+1,J))) *
+       (TS(I+1,J)-TS(I,J))/DX(J)
C  GRADIENTE TEMPERATURA REFERENCIA
        TENDU(I,J) = TENDU(I,J)
+       + G*(0.5*(E1(I,J) + E1(I+1,J)) + (E2(I,J) + E2(I+1,J)) *
+       (0.5*(TH(I,J) + TH(I+1,J))-0.5*(TR(I,J) + TR(I+1,J))))
+       /(GAMMA*(TR(I,J) + TR(I+1,J)))
+       *(TR(I+1,J)-TR(I,J))/DX(J)
C  GRADIENTE TOPOGRAFIA
        TENDU(I,J) = TENDU(I,J)
+       -G*(GAMMA*0.5*(E1(I,J) + E1(I+1,J))-0.5*(E2(I,J) + E2(I+1,J))
+       + 1.)*(ZS(I+1,J)-ZS(I,J))/DX(J)
C  GRADIENTE ALTURA DE REFERENCIA
        TENDU(I,J) = TENDU(I,J)
+       -G*(0.5*(E2(I,J) + E2(I+1,J))-0.5*(E1(I,J) + E1(I+1,J))*GAMMA)
+       *(ZR(I+1,J)-ZR(I,J))/DX(J)
C  FRICCION SUPERFICIAL
        TENDU(I,J) = TENDU(I,J)
+       -CD(I,J)*CROZ*2.8*U(I,J)*SQRT(ABS(U(I,J))**2. +
+       (ABS(.25*(V(I,J) + V(I,J-1) + V(I+1,J) + V(I+1,J-1))))**2.)/H
C  DIFUSION EN NIVEL SIGMA DE COMPONENTE ZONAL
        TENDU(I,J) = TENDU(I,J) + KM*A(I,J)
C  CALCULO TENDENCIA DE LA COMPONENTE MERIDIANA DEL VIENTO EN EL SUELO
C
C  ADVECCION COMPONENTE MERIDIANA
        TENDV(I,J) = -0.25*(U(I,J) + U(I-1,J) + U(I,J+1) + U(I-1,J+1)) *
+       (V(I+1,J)-V(I-1,J))/(2.*DX(J))
        TENDV(I,J) = TENDV(I,J)
+       -V(I,J)*(V(I,J+1)-V(I,J-1))/(2.*DY)

```

```

C  TERMINO DE CORIOLIS
  IF(CORIOL) TENDV(I,J) = TENDV(I,J)
  +          -0.5*(COR(J) + COR(J+1))*0.25*
  +          (U(I,J) + U(I-1,J) + U(I,J+1) + U(I-1,J+1))
C  GRADIENTE TEMPERATURA SUPERFICIE
  TENDV(I,J) = TENDV(I,J)
  +          -G*0.5*((E1(I,J)-H/TH(I,J)) + (E1(I,J+1)-H/TH(I,J+1))) *
  +          (TS(I,J+1)-TS(I,J))/DY
C  GRADIENTE TEMPERATURA REFERENCIA
  TENDV(I,J) = TENDV(I,J)
  +          + G*(0.5*(E1(I,J+1) + E1(I,J)) + (E2(I,J+1) + E2(I,J)) *
  +          (0.5*(TH(I,J+1) + TH(I,J)) - 0.5*(TR(I,J+1) + TR(I,J)))
  +          /(GAMMA*(TR(I,J+1) + TR(I,J))))
  +          *((TR(I,J+1)-TR(I,J))/DY)
C  GRADIENTE TOPOGRAFIA
  TENDV(I,J) = TENDV(I,J)
  +          -G*(GAMMA*0.5*(E1(I,J) + E1(I,J+1)) - 0.5*(E2(I,J) + E2(I,J+1))
  +          + 1.)*(ZS(I,J+1)-ZS(I,J))/DY
C  GRADIENTE ALTURA DE REFERENCIA
  TENDV(I,J) = TENDV(I,J)
  +          -G*(0.5*(E2(I,J) + E2(I,J+1)) - 0.5*(E1(I,J) + E1(I,J+1))*GAMMA)
  +          *(ZR(I,J+1)-ZR(I,J))/DY
C  FRICCION SUPERFICIAL
  TENDV(I,J) = TENDV(I,J)
  +          -CD(I,J)*CROZ*2.8*V(I,J)*SQRT(ABS(V(I,J))**2. +
  +          (ABS(.25*(U(I,J) + U(I-1,J) + U(I,J+1) + U(I-1,J+1))))**2.)/H
C  DIFUSION EN NIVEL SIGMA DE COMPONENTE ZONAL
  TENDV(I,J) = TENDV(I,J) + KM*C(I,J)
210  CONTINUE
  TNDTEM = 0.
  TNDCOU = 0.
  TNDCOV = 0.
  DIFTEM = 0.
  DIFCOU = 0.
  DIFCOV = 0.
  DO 250 I = 1, LG
    DO 250 J = 1, MG
      TNDTEM = TNDTEM + TENDT(I,J)
      TNDCOU = TNDCOU + TENDU(I,J)
      TNDCOV = TNDCOV + TENDV(I,J)
      DIFTEM = DIFTEM + KT*B(I,J)/A1(I,J)
      DIFCOU = DIFCOU + KM*A(I,J)
      DIFCOV = DIFCOV + KM*C(I,J)
250  CONTINUE
  TNDTEM = TNDTEM/(LG*MG)
  TNDCOU = TNDCOU/(LG*MG)
  TNDCOV = TNDCOV/(LG*MG)
  DIFTEM = DIFTEM/(LG*MG)
  DIFCOU = DIFCOU/(LG*MG)
  DIFCOV = DIFCOV/(LG*MG)
251  FORMAT(I7,4X,3(F8.6,4X))
  RETURN
  END

```

SUBROUTINE BOUND

```

C
*****
*
* FUNCTION * BOUNDARY CONDITIONS.
*
*****
C
PARAMETER(LG = 54,MG = 54)
C
COMMON /A01/ L,      M,      L1,      M1,      M2,      G,
1      SLAT, DPHI
COMMON /A02/ NSTEP, DT, DUSMIN, ALFA(LG,MG),ISTEP,NADIAB
COMMON /A03/ DX(MG), COR(MG), DY
COMMON /A04/ ICAB(30),TNDTEM,TNDCOU,TNDCOV
COMMON /A05/ ZS(LG,MG), ZR(LG,MG), U(LG,MG), V(LG,MG),
1      TR(LG,MG), TS(LG,MG), TH(LG,MG), CD(LG,MG),
1      AC(LG,MG)
COMMON /A06/ A(LG,MG), B(LG,MG), C(LG,MG)
COMMON /A07/ A1(LG,MG), A2(LG,MG), A3(LG,MG), C1(LG,MG),
1      C2(LG,MG), E1(LG,MG), E2(LG,MG)
COMMON /A08/ GAMMAD, GAMMA, CP,
1      KM, KT, H, RNU, SMOROG, FACTOR, OROMAX,TSUP,
1      CDSEA,CDLAND,CROZ,ACSEA,ACLAND,VGEOST,EPSI
COMMON /A09/ TENDT(LG,MG),TENDU(LG,MG),TENDV(LG,MG)
COMMON /A10/ TENDT1(LG,MG),TENDU1(LG,MG),TENDV1(LG,MG)
COMMON /A11/ BOUNDT(LG,MG),BOUNDU(LG,MG),BOUNDV(LG,MG)
COMMON /GRAF/ LIMX,LIMY,MAX_COLOR,MODE,CORIOI
LOGICAL TSUP,VGEOST
REAL KT,KM

DO 301 J = 1,MG
  IF(I.EQ.1.AND.U(1,J).LT.0.) THEN
    TS(1,J) = TS(2,J)
    U(1,J) = U(2,J)
    V(1,J) = V(2,J)
  ELSEIF(I.EQ.LG.AND.U(LG,J).GT.0.) THEN
    TS(LG,J) = TS(LG-1,J)
    U(LG,J) = U(LG-1,J)
    V(LG,J) = V(LG-1,J)
  ENDIF
301 CONTINUE
DO 302 I = 1,LG
  IF(J.EQ.1.AND.V(I,1).LT.0.) THEN
    TS(I,1) = TS(I,2)
    U(I,1) = U(I,2)
    V(I,1) = V(I,2)
  ELSEIF(J.EQ.MG.AND.V(I,MG).GT.0.) THEN
    TS(I,MG) = TS(I,MG-1)
    U(I,MG) = U(I,MG-1)
    V(I,MG) = V(I,MG-1)
  ENDIF
302 CONTINUE
RETURN
END

```

SUBROUTINE TSTEP

```

C
*****
*
* FUNCTION * TIME INTEGRATION SCHEME: MODIFIED SECOND ORDER
*           * ADAM'S BASHFORTH SCHEME.
*           * THIS ROUTINE RELAXES THE NEW FORECASTED VALUES
*           * TO THE PRESCRIBED INITIAL BOUNDARY VALUES (KALLBERG-
*           * DAVIES SCHEME).
*
*****
C
PARAMETER(LG = 54,MG = 54)
C
COMMON /A01/ L,      M,      L1,      M1,      M2,      G,
1      SLAT, DPHI
COMMON /A02/ NSTEP, DT,  DUSMIN, ALFA(LG,MG),ISTEP,NADIAB
COMMON /A03/ DX(MG),   COR(MG),   DY
COMMON /A04/ ICAB(30),TNDTEM,TNDCOU,TNDCOV
COMMON /A05/ ZS(LG,MG), ZR(LG,MG), U(LG,MG), V(LG,MG),
1      TR(LG,MG), TS(LG,MG), TH(LG,MG), CD(LG,MG),
1      AC(LG,MG)
COMMON /A06/ A(LG,MG),   B(LG,MG),   C(LG,MG)
COMMON /A07/ A1(LG,MG),  A2(LG,MG),  A3(LG,MG),  C1(LG,MG),
1      C2(LG,MG), E1(LG,MG), E2(LG,MG)
COMMON /A08/ GAMMAD, GAMMA, CP,
1      KM, KT, H, RNU, SMOROG, FACTOR, OROMAX,TSUP,
1      CDSEA,CDLAND,CROZ,ACSEA,ACLAND,VGEOST,EPSI
COMMON /A09/ TENDT(LG,MG),TENDU(LG,MG),TENDV(LG,MG)
COMMON /A10/ TENDT1(LG,MG),TENDU1(LG,MG),TENDV1(LG,MG)
COMMON /A11/ BOUNDT(LG,MG),BOUNDU(LG,MG),BOUNDV(LG,MG)
COMMON /GRAF/ LIMX,LIMY,MAX_COLOR,MODE,CORIOL
REAL KT,KM
LOGICAL TSUP,VGEOST
IF(ISTEP.EQ.1) THEN
  DO 300 J = 2,MG-1
    DO 300 I = 2,LG-1
      TS(I,J) = (TS(I,J) + TENDT(I,J)*DT)
      U(I,J) = (U(I,J) + TENDU(I,J)*DT)
      V(I,J) = (V(I,J) + TENDV(I,J)*DT)
300 CONTINUE
ELSE
C
C MODIFIED SECOND-ORDER ADAM'S BASHFORTH TIME SCHEME,
C AND BOUNDARY RELAXATION (DAVIES, 1976).
C
DO 310 J = 2,MG-1
DO 310 I = 2,LG-1
  TS(I,J) = (TS(I,J) + (3.*TENDT(I,J)-TENDT1(I,J))*DT*0.5)
  U(I,J) = (U(I,J) + (3.*TENDU(I,J)-TENDU1(I,J))*DT*0.5)
  V(I,J) = (V(I,J) + (3.*TENDV(I,J)-TENDV1(I,J))*DT*0.5)
310 CONTINUE
ENDIF

```



```

CALL SMOOTH(TS,B,LG,MG,RNU)
CALL SMOOTH(U,B,LG,MG,RNU)
CALL SMOOTH(V,B,LG,MG,RNU)
  DO 320 J=1,MG
    DO 320 I=1,LG
      TENDT1(I,J)=TENDT(I,J)
      TENDU1(I,J)=TENDU(I,J)
      TENDV1(I,J)=TENDV(I,J)
320 CONTINUE
  TS0=0.
  U0=0.
  V0=0.
  DO 321 I=1,LG
    DO 321 J=1,MG
      TS0=TS0+TS(I,J)
      U0=U0+U(I,J)
      V0=V0+V(I,J)
321 CONTINUE
  TS0=TS0/(LG*MG)
  U0=U0/(LG*MG)
  V0=V0/(LG*MG)
322 FORMAT(30X,I7,4X,3(F8.2,4X))
  RETURN
  END

```

SUBROUTINE LAPSIG(A,B,DX,DY,L,M)

```
*****
*                               *
* FUNCTION * SUBROUTINE 'LAPSIG' CALCULATES THE LAPLACIAN OF ANY *
*          * SCALAR FIELD *
*          *
* SYMBOLS * A(L,M) LAPLACIAN OF B. *
* USED    *
*          * B(L,M) ANY SCALAR FIELD. *
*          * CORRESPONDS TO THE STREAMFUNCTION FIELD. *
*          *
*          * DX(M) GRID DISTANCE IN METERS IN THE ZONAL *
*          * DIRECTION. *
*          * DY GRID DISTANCE IN METERS IN THE *
*          * MERIDIONAL DIRECTION. *
*          * L NUMBER OF GRID POINTS IN THE ZONAL *
*          * DIRECTION. *
*          * M NUMBER OF GRID POINTS IN THE MERIDIONAL *
*          * DIRECTION. *
*****
```

C

DIMENSION A(L,M), B(L,M), DX(M)

C

C**A

C

M1=M-1

M2=M-2

L1=L-1

L2=L-2

DO 95 J=2,M1

JP1=J+1

JM1=J-1

DO 95 I=2,L1

95 A(I,J) = ((B(I+1,J) + B(I-1,J) - 2.*B(I,J))/DX(J)**2
1 + (B(I,JP1) + B(I,JM1) - 2.*B(I,J))/DY**2)

C

C**B

C

DO 96 J=2,M1

A(1,J) = ((B(2,J) - B(1,J))/DX(J)**2

1 + (B(1,J+1) + B(1,J-1) - 2.*B(1,J))/DY**2)

A(L,J) = ((B(L1,J) - B(L,J))/DX(J)**2

1 + (B(L,J+1) + B(L,J-1) - 2.*B(L,J))/DY**2)

96 CONTINUE

DO 97 I=2,L1

A(I,1) = ((B(I,2) - B(I,1))/DY**2

1 + (B(I+1,1) + B(I-1,1) - 2.*B(I,1))/DX(1)**2)

A(I,M) = ((B(I,M1) - B(I,M))/DY**2

1 + (B(I+1,M) + B(I-1,M) - 2.*B(I,M))/DX(M)**2)

97 CONTINUE

A(1,1) = (B(2,1) - B(1,1))/DX(1)**2 + (B(1,2) - B(1,1))/DY**2

A(L,1) = (B(L1,1) - B(L,1))/DX(1)**2 + (B(L,2) - B(L,1))/DY**2

A(1,M) = (B(2,M) - B(1,M))/DX(M)**2 + (B(1,M1) - B(1,M))/DY**2

A(L,M) = (B(L1,M) - B(L,M))/DX(M)**2 + (B(L,M1) - B(L,M))/DY**2

C

RETURN

END

```

SUBROUTINE DIBWIN(U,V,ZS,IX,IY,SIZE,ANCHO,ISTEP,ISKIPP,IERR)
COMMON /GRAF/ LIMX,LIMY,MAX_COLOR,MODE,ICON,CORIOL
COMMON /A01/ L, M, L1, M1, M2, G,
1 SLAT, DPHI
COMMON /A08/ GAMMAD, GAMMA, CP,
1 KM, KT, H, RNU, SMOROG, FACTOR, OROMAX,TSUP,
1 CDSEA,CDLAND,CROZ,ACSEA,ACLAND,VGEOST,EPSI
COMMON /A12/ DIA,HORA,ALCAN,REFLEV
DIMENSION U(IX,IY),V(IX,IY),ZS(IX,IY)
INCLUDE 'GREX.FH'
CHARACTER*2 DIA,ALCAN
CHARACTER*1 HORA
CHARACTER*4 PASO
CHARACTER*6 ROZA, FACT, PSI, CSEA, CLAND, ASEA, ALAND
CHARACTER*10 OROMX, KMD, KTD, GAMM
CHARACTER*4 CORI, VGT, TSP
CHARACTER*64 STRING
WRITE(PASO,'(I4)')ISTEP
WRITE(ROZA,'(F6.2)')CROZ
WRITE(FACT,'(F6.2)')FACTOR
WRITE(PSI,'(F6.2)')EPSI
WRITE(GAMM,'(F10.4)')GAMMA
WRITE(CSEA,'(F6.4)')CDSEA
WRITE(CLAND,'(F6.4)')CDLAND
WRITE(ASEA,'(F6.4)')ACSEA
WRITE(ALAND,'(F6.4)')ACLAND
WRITE(OROMX,'(F10.2)')OROMAX
WRITE(KMD,'(F10.2)')KM
WRITE(KTD,'(F10.2)')KT
WRITE(CORI,'(L4)')CORIOL
WRITE(VGT,'(L4)')VGEOST
WRITE(TSP,'(L4)')TSUP
STRING = "VIENTO INICIAL "
ICON=ICON+1
IF(ICON.GT.1) THEN
STRING = "VIENTO ADAPTADO"
ENDIF

```

```

IER=GRAPHIC_TEXT('DIA = ',450,50,0)
IER=GRAPHIC_TEXT(DIA,500,50,0)
IER=GRAPHIC_TEXT('HORA = ',450,80,0)
IER=GRAPHIC_TEXT(HORA,500,80,0)
IER=GRAPHIC_TEXT('ALCAN = ',450,110,0)
IER=GRAPHIC_TEXT(ALCAN,520,110,0)
IER=GRAPHIC_TEXT('CORIOL = ',450,140,0)
IER=GRAPHIC_TEXT(CORI,500,140,0)
IER=GRAPHIC_TEXT('VGEOST = ',450,170,0)
IER=GRAPHIC_TEXT(VGT,500,170,0)
IER=GRAPHIC_TEXT('TSUP = ',450,200,0)
IER=GRAPHIC_TEXT(TSP,500,200,0)
IER=GRAPHIC_TEXT('CROZ = ',450,230,0)
IER=GRAPHIC_TEXT(ROZA,500,230,0)
C IER=GRAPHIC_TEXT('FACT = ',120,80,15)
C IER=GRAPHIC_TEXT(FACT,170,80,15)
C IER=GRAPHIC_TEXT('EPSI = ',240,80,15)
C IER=GRAPHIC_TEXT(PSI,290,80,15)
IER=GRAPHIC_TEXT('GAMMA = ',450,260,0)
IER=GRAPHIC_TEXT(GAMM,500,260,0)

```

```

IER=GRAPHIC_TEXT('OROMAX = ',450,290,0)
IER=GRAPHIC_TEXT(OROMX,500,290,0)
C IER=GRAPHIC_TEXT('KM = ',150,110,15)
C IER=GRAPHIC_TEXT(KMD,190,110,15)
C IER=GRAPHIC_TEXT('KT = ',300,110,15)
C IER=GRAPHIC_TEXT(KTD,350,110,15)
C IER=GRAPHIC_TEXT('CDSEA = ',5,140,15)
C IER=GRAPHIC_TEXT(CSEA,80,140,15)
IER=GRAPHIC_TEXT('CDLAND = ',450,320,0)
IER=GRAPHIC_TEXT(CLAND,530,320,0)
C IER=GRAPHIC_TEXT('ACSEA = ',330,140,15)
C IER=GRAPHIC_TEXT(ASEA,400,140,15)
C IER=GRAPHIC_TEXT('ACLAND = ',470,140,15)
C IER=GRAPHIC_TEXT(ALAND,550,140,15)

C
C RECUADRO
C
IER=SET_COLOR(0)
IM=0
IN=LIMY-20
IER=MOVE(IM,IN)
IER=DRAW(IM+(IX+1)*INT(ANCHO),IN)
IER=DRAW(IM+(IX+1)*INT(ANCHO),IN-(IY+1)*INT(ANCHO))
IER=DRAW(IM,IN-(IY+1)*INT(ANCHO))
IER=DRAW(IM,IN)
IER=GRAPHIC_TEXT(STRING,IM,IN-(2+IY)*INT(ANCHO),0)
IER=GRAPHIC_TEXT('25 kt',INT(ANCHO*IX/2.)
+ ,IN-(2+IY)*INT(ANCHO),0)
IER=GRAPHIC_TEXT(PASO,IM+(IX-4)*INT(ANCHO)
+ ,IN-(2+IY)*INT(ANCHO),0)

UR=12.5
VR=0.
FUERZA=UR
IR=INT(ANCHO*3*IX/8.)
JR=IN-(2+IY)*INT(ANCHO)
IER=MOVE(IR,JR)
IR1=IR+INT(UR*SIZE)
JR1=JR-INT(VR*SIZE)
IER=DRAW(IR1,JR1)
IER=MOVE(IR1,JR1)
IER=FILLED_RECTANGLE(IR1-1,JR1-1,IR1+1,JR1+1)
IER=MOVE(IR,JR)
IR1=IR-INT(UR*SIZE/8)
JR1=JR-INT(UR*SIZE/4)
IER=DRAW(IR1,JR1)
IR1=IR+INT(UR*SIZE/5.)
JR1=JR
IER=MOVE(IR1,JR1)
IR1=IR1-INT(UR*SIZE/8)
JR1=JR1-INT(UR*SIZE/4)
IER=DRAW(IR1,JR1)
IR1=IR+INT(UR*2*SIZE/5.)
JR1=JR
IER=MOVE(IR1,JR1)
IR1=IR1-INT(UR*SIZE/16)
JR1=JR1-INT(UR*SIZE/8)
IER=DRAW(IR1,JR1)

```

```

DO 300 J=1,IY
  DO 300 I=1,IX
    IM=INT(I*ANCHO)
    IN=LIMY-INT(J*ANCHO)-20
    IER=MOVE(IM,IN)
    IF((ZS(I,J).GT.1.0).AND.(ZS(I,J).LE.500.0)) THEN
      IER=SET_COLOR(14)
      IER=FILLED_RECTANGLE(IM-5,IN-5,IM+5,IN+5)
    ELSEIF((ZS(I,J).GT.500.0).AND.(ZS(I,J).LE.1000.0)) THEN
      IER=SET_COLOR(13)
      IER=FILLED_RECTANGLE(IM-5,IN-5,IM+5,IN+5)
    ELSEIF((ZS(I,J).GT.1000.0).AND.(ZS(I,J).LE.1500.0)) THEN
      IER=SET_COLOR(12)
      IER=FILLED_RECTANGLE(IM-5,IN-5,IM+5,IN+5)
    ELSEIF((ZS(I,J).GT.1500.0).AND.(ZS(I,J).LE.2000.0)) THEN
      IER=SET_COLOR(11)
      IER=FILLED_RECTANGLE(IM-5,IN-5,IM+5,IN+5)
    ELSEIF(ZS(I,J).GT.2000.0) THEN
      IER=SET_COLOR(10)
      IER=FILLED_RECTANGLE(IM-5,IN-5,IM+5,IN+5)
    ELSE
      CONTINUE
      IER=SET_COLOR(15)
      IER=FILLED_RECTANGLE(IM-5,IN-5,IM+5,IN+5)
    ENDIF
  300 CONTINUE
C
C VIENTOS
C
  DO 200 J=1,IY,ISKIPP
    DO 200 I=1,IX,ISKIPP
      IM=INT(I*ANCHO)
      IN=LIMY-INT(J*ANCHO)-20
      IF(ZS(I,J).LT.1.0) THEN
        IER=SET_COLOR(0)
        FUERZA=SQRT(U(I,J)**2.+V(I,J)**2.)
C      PINTO EL BRAZO DEL VIENTO
        IER=MOVE(IM,IN)
        IER=FILLED_ELLIPSE(IM,IN,1,1)
C      BRAZOS DE LOS VIENTOS DE IGUAL LONGUTUD
        IU=IM-INT((U(I,J)*SIZE*20.*ISKIPP)/(FUERZA*5.))
        IV=IN+INT((V(I,J)*SIZE*20.*ISKIPP)/(FUERZA*5.))
C      BRAZOS DE LOS VIENTOS PROPORCIONALES AL VIENTO
C      IU=IM-INT(U(I,J)*SIZE/1.5)
C      IV=IN+INT(V(I,J)*SIZE/1.5)
        IER=DRAW(IU,IV)
C      DISCRIMINO LOS VALORES DEL VIENTO PARA PINTAR LAS BARBAS
C      NOTA: EL VALOR MAXIMO DE VIENTO QUE REPRESENTO ES 70 kt
        DIS=SQRT((IU-IM)**2.+(IV-IN)**2.)
        IF(FUERZA.GT.24.0) THEN
          IER=MOVE(IU,IV)
          IR1=IU-INT((IV-IN)*SIZE/DIS*4)
          JR1=IV-INT((IM-IU)*SIZE/DIS*4)
          IER=DRAW(IR1,JR1)
          IER=MOVE(IU+((IM-IU)/10),IV+((IN-IV)/10))
          IER=DRAW(IR1,JR1)
          IER=MOVE(IU+((IM-IU)/5),IV+((IN-IV)/5))
          IER=DRAW(IR1,JR1)

```

```

IER = MOVE(IU + ((IM-IU)/10*3),IV + ((IN-IV)/10*3))
IER = DRAW(IR1,JR1)
IER = MOVE(IU + ((IM-IU)/5*2),IV + ((IN-IV)/5*2))
IER = DRAW(IR1,JR1)
  IF(FUERZA.GT.28.0) THEN
    IER = MOVE(IU + ((IM-IU)/10*5),IV + ((IN-IV)/10*5))
    IR1 = IU + ((IM-IU)/10*5)-INT((IV-IN)*SIZE/DIS*4)
    JR1 = IV + ((IN-IV)/10*5)-INT((IM-IU)*SIZE/DIS*4)
    IER = DRAW(IR1,JR1)
  ENDIF
  IF(FUERZA.GT.33.0) THEN
    IER = MOVE(IU + ((IM-IU)/5*3),IV + ((IN-IV)/5*3))
    IR1 = IU + ((IM-IU)/5*3)-INT((IV-IN)*SIZE/DIS*4)
    JR1 = IV + ((IN-IV)/5*3)-INT((IM-IU)*SIZE/DIS*4)
    IER = DRAW(IR1,JR1)
  ENDIF
  GOTO 200
ENDIF
IF(FUERZA.GT.2.0) THEN
  IER = MOVE(IU,IV)
  IR1 = IU-INT((IV-IN)*SIZE/DIS*2)
  JR1 = IV-INT((IM-IU)*SIZE/DIS*2)
  IER = DRAW(IR1,JR1)
ENDIF
IF(FUERZA.GT.4.0) THEN
  IER = MOVE(IU,IV)
  IR1 = IU-INT((IV-IN)*SIZE/DIS*4)
  JR1 = IV-INT((IM-IU)*SIZE/DIS*4)
  IER = DRAW(IR1,JR1)
ENDIF
IF(FUERZA.GT.6.5) THEN
  IER = MOVE(IU + ((IM-IU)/5),IV + ((IN-IV)/5))
  IR1 = IU + ((IM-IU)/5)-INT((IV-IN)*SIZE/DIS*2)
  JR1 = IV + ((IN-IV)/5)-INT((IM-IU)*SIZE/DIS*2)
  IER = DRAW(IR1,JR1)
ENDIF
IF(FUERZA.GT.9.0) THEN
  IER = MOVE(IU + ((IM-IU)/5),IV + ((IN-IV)/5))
  IR1 = IU + ((IM-IU)/5)-INT((IV-IN)*SIZE/DIS*4)
  JR1 = IV + ((IN-IV)/5)-INT((IM-IU)*SIZE/DIS*4)
  IER = DRAW(IR1,JR1)
ENDIF
IF(FUERZA.GT.11.5) THEN
  IER = MOVE(IU + ((IM-IU)/5*2),IV + ((IN-IV)/5*2))
  IR1 = IU + ((IM-IU)/5*2)-INT((IV-IN)*SIZE/DIS*2)
  JR1 = IV + ((IN-IV)/5*2)-INT((IM-IU)*SIZE/DIS*2)
  IER = DRAW(IR1,JR1)
ENDIF
IF(FUERZA.GT.14.0) THEN
  IER = MOVE(IU + ((IM-IU)/5*2),IV + ((IN-IV)/5*2))
  IR1 = IU + ((IM-IU)/5*2)-INT((IV-IN)*SIZE/DIS*4)
  JR1 = IV + ((IN-IV)/5*2)-INT((IM-IU)*SIZE/DIS*4)
  IER = DRAW(IR1,JR1)
ENDIF
IF(FUERZA.GT.16.5) THEN
  IER = MOVE(IU + ((IM-IU)/5*3),IV + ((IN-IV)/5*3))
  IR1 = IU + ((IM-IU)/5*3)-INT((IV-IN)*SIZE/DIS*2)
  JR1 = IV + ((IN-IV)/5*3)-INT((IM-IU)*SIZE/DIS*2)

```

```

    IER = DRAW(IR1,JR1)
ENDIF
IF(FUERZA.GT.19.0) THEN
    IER = MOVE(IU + ((IM-IU)/5*3),IV + ((IN-IV)/5*3))
    IR1 = IU + ((IM-IU)/5*3)-INT((IV-IN)*SIZE/DIS*4)
    JR1 = IV + ((IN-IV)/5*3)-INT((IM-IU)*SIZE/DIS*4)
    IER = DRAW(IR1,JR1)
ENDIF
IF(FUERZA.GT.21.5) THEN
    IER = MOVE(IU + ((IM-IU)/5*4),IV + ((IN-IV)/5*4))
    IR1 = IU + ((IM-IU)/5*4)-INT((IV-IN)*SIZE/DIS*2)
    JR1 = IV + ((IN-IV)/5*4)-INT((IM-IU)*SIZE/DIS*2)
    IER = DRAW(IR1,JR1)
ENDIF
ENDIF
200 CONTINUE
    IER = SET_COLOR(0)
    IER = GRAPHIC_TEXT('PRESS A KEY TO CONTINUE',10,LIMY-10,0)
    IER = PAUSE()
    RETURN
END

```



```
SUBROUTINE IBUF(IU, FIELD, IX, IY, ELEM, IERR)
  DIMENSION FIELD(IX, IY)
  COMMON /GRAF/ LIMX, LIMY, MAX_COLOR, MODE, CORIOL
  CHARACTER*10 CONTOR
  CHARACTER*22 FICH
  CHARACTER*8 ELEM
  CHARACTER*4 EXT
  DATA EXT/' .DAT'/
  DATA CONTOR/'C:\BASURA\'/
  FICH=CONTOR//ELEM//EXT
  PRINT*, ' ABRE ', FICH
  OPEN(IU, FILE=FICH)
  DO 110 J=54, 1, -1
    READ(IU, 100)(FIELD(I, J), I=1, IX)
110  CONTINUE
100  FORMAT(6(9(F7.2, 1X)/))
  CLOSE(IU)
  RETURN
  END
```

```

SUBROUTINE SMOOTH(A,B,L,M,RNU)
C
C THIS SUBROUTINE PERFORMS THE NINE-POINT SMOOTHING OF
C THE ANALYZED FIELD. 'A' IS THE VALUE TO BE SMOOTHED
C WHILE 'B' IS THE SMOOTHED PARAMETER PASSED BACK.
C
  DIMENSION A(L,M),B(L,M)
  L1 = L-1
  M1 = M-1
C
C COMPUTATION OF THE SMOOTHING COEFFICIENTS.
C
  RNU = 0.2
  RNU1 = 0.5*RNU*(1.-RNU)
  RNU2 = 0.25*RNU*RNU
C
  DO 10 I=2,L1
    IP = I+1
    IM = I-1
C
C COMPUTATION OF THE SMOOTHED COEFFICIENTS.
C
    DO 10 J=2,M1
      B(I,J)=A(I,J)+RNU1*(A(IM,J)+A(IP,J)+A(I,J+1)+
1 A(I,J-1)-4.*A(I,J))
2 +RNU2*(A(IM,J+1)+A(IM,J-1)+A(IP,J+1)+
3 A(IP,J-1)-4.*A(I,J))
10 CONTINUE
    DO 11 I=2,L1
      DO 11 J=2,M1
        A(I,J) = B(I,J)
11 CONTINUE
    RETURN
  END

```

```

subroutine show_video_systems
integer video_system(4),video_configuration,mode,card,crt

mode = video_configuration(video_system)

card = video_system(1)
crt = video_system(2)

if(card.eq.0) then
    write(6,*) ' No video adapter present'
    go to 86
endif

write(6,101) ' Active: '

if(card.eq. 1) write(6,101) 'MDA: Monochrome Display Adapter'
if(card.eq. 2) write(6,101) 'CGA: Color Graphics Adapter'
if(card.eq. 3) write(6,101) 'EGA: Enhanced Graphics Adapter'
if(card.eq. 4) write(6,101) 'MCGA: Multi Color Gate Array'
if(card.eq. 5) write(6,101) 'VGA: Video Graphics Array'
if(card.eq.64) write(6,101) 'HGC: Hercules Graphics Card'
101 format(1x,a,$)
if(crt.eq.1) write(6,*) ' with MDA-compatible monochrome display'
if(crt.eq.2) write(6,*) ' with CGA-compatible color display'
if(crt.eq.3) write(6,*) ' with EGA-compatible color display'
if(crt.eq.4) write(6,*) ' with PS/2-compatible monochrome display'
if(crt.eq.5) write(6,*) ' with PS/2-compatible color display'

card = video_system(3)
crt = video_system(4)

write(6,101) ' Inactive: '

if(card.eq.0) write(6,*) 'Not installed.'

if(card.eq. 1) write(6,101) 'MDA: Monochrome Display Adapter'
if(card.eq. 2) write(6,101) 'CGA: Color Graphics Adapter'
if(card.eq. 3) write(6,101) 'EGA: Enhanced Graphics Adapter'
if(card.eq. 4) write(6,101) 'MCGA: Multi Color Gate Array'
if(card.eq. 5) write(6,101) 'VGA: Video Graphics Array'
if(card.eq.64) write(6,101) 'HGC: Hercules Graphics Card'

if(crt.eq.1) write(6,*) ' with MDA-compatible monochrome display'
if(crt.eq.2) write(6,*) ' with CGA-compatible color display'
if(crt.eq.3) write(6,*) ' with EGA-compatible color display'
if(crt.eq.4) write(6,*) ' with PS/2-compatible monochrome display'
if(crt.eq.5) write(6,*) ' with PS/2-compatible color display'

86 return
end

```

C-----

```

character function crawl(x,y)
include 'grex.fh'
integer xmin,ymin,xmax,ymax

```

```

integer k,j,x,y,step
character q,nul,cv*13
ier = get_clip_limits(xmin,ymin,xmax,ymax)
ix = xmax - 98
jx = xmax - 2
iy = ymax - 10
jy = ymax - 2
nul = char(0)
step = 4
10 ier = move_cursor(x,y)
   j = get_pixel(x,y)
   write(cv,301) x,y,j,nul
301 format(3i4,a)
   call set_color(0)
   call filled_rectangle(ix,iy,jx,jy)
   call graphic_text(cv,ix,iy,9)
   k = pause()
   if(k.lt.0) then
      k = -k
      if(k.eq.83) step = step-1
      if(k.eq.82) step = step+1
      step = max0(step,1)
      step = min0(step,100)
      if(k.lt.71.or.k.gt.81) go to 10
      if(k.eq.71.or.k.eq.72.or.k.eq.73) y = y - step
      if(k.eq.79.or.k.eq.80.or.k.eq.81) y = y + step
      if(k.eq.71.or.k.eq.75.or.k.eq.79) x = x - step
      if(k.eq.73.or.k.eq.77.or.k.eq.81) x = x + step
      if(x.lt.xmin) x = xmax + x - xmin + 1
      if(y.lt.ymin) y = ymax + y - ymin + 1
      if(x.gt.xmax) x = xmax - x + xmin + 1
      if(y.gt.ymax) y = ymax - y + ymin + 1
      go to 10
   endif
   crawl = char(k)
   ier = move_cursor(-1,-1)
   call set_color(0)
   call filled_rectangle(ix,iy,jx,jy)
   return
end

```

C-----

```

integer function get_string(x,y,prompt,string)
c
c Routine to get a string from the user interactively, displaying
c the results on screen, and allowing the user to backspace over
c mistakes. Return or line feed terminate the procedure.
c
integer x,y
integer ix,iy,key,ic,ier
integer xmin,ymin,xmax,ymax
character*(*) prompt,string
character nul
include 'gregx.fh'
c
c nul value
c nul = char(0)
)

```

```

c
c  write prompt to screen
  ix = x
  iy = y
  call graphic_text(prompt,ix,iy,14)
c
c  find last nonblank character in prompt
  j = 1 + len(prompt)
1 j = j-1
  if(j.le.0) go to 2
  if(prompt(j:j).eq.' '.or.prompt(j:j).eq.nul) go to 1
2 j = max0(j,0)
c
c  adjust ix to describe position of the last nonblank character in prompt
  ix = ix + j*8
c
c  add one extra space
  ix = ix + 8
c
c  compare ix with clip limits; clip to within limits
  call get_clip_limits(xmin,ymin,xmax,ymax)
  ix = min0(ix,xmax)
c
c  initialize string and set up a character counter
  string = ''
  ic = 0
c
c  While the user has not entered a return or a line feed
c  get a key from user
5 key = pause()
  if(key.le.0) go to 5
  if(key.eq.13) go to 86
  if(key.eq.10) go to 86
  if(key.eq.8) then

c      erase a character only if one is there to erase
      if(ic.gt.0) then
c          adjust character position pointer
          ix = ix - 8
c          erase character from screen using xor function
          call set_xor(1)
          call graphic_text(string(ic:ic),ix,iy,10)
          call set_xor(0)
c          remove character from string
          string(ic:ic) = ''
          ic = ic-1
      endif

      else

c          write character on screen
          call graphic_text(char(key),ix,iy,10)
c          advance to next position
          ix = ix + 8
c          store character in string
          ic = ic+1
c          if there is room
          ic = min0(ic,len(string))

```

```
c      otherwise replace last character of string
      string(ic:ic) = char(key)
```

```
      endif
```

```
      go to 5
```

```
c      End-while
```

```
86 return
```

```
end
```

```
integer function actual_length (s)
```

```
character*(*) s
```

```
j = 1 + len(s)
```

```
1 j = j-1
```

```
if(j.le.0) go to 2
```

```
if(s(j:j).eq.' ') go to 1
```

```
2 actual_length = j
```

```
return
```

```
end
```

```
C-----
```